

University of Southampton Research Repository ePrints Soton

Copyright © and Moral Rights for this thesis are retained by the author and/or other copyright owners. A copy can be downloaded for personal non-commercial research or study, without prior permission or charge. This thesis cannot be reproduced or quoted extensively from without first obtaining permission in writing from the copyright holder/s. The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the copyright holders.

When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given e.g.

AUTHOR (year of submission) "Full thesis title", University of Southampton, name of the University School or Department, PhD Thesis, pagination

UNIVERSITY OF SOUTHAMPTON

Automatic Image Annotation and Object Detection

by

Jiayu Tang

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
Faculty of Engineering, Science and Mathematics
School of Electronics and Computer Science

May 2008

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING, SCIENCE AND MATHEMATICS
SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

by Jiayu Tang

We live in the midst of the information era, during which organising and indexing information more effectively is a matter of essential importance. With the fast development of digital imagery, how to search images - a rich form of information - more efficiently by their content has become one of the biggest challenges.

Content-based image retrieval (CBIR) has been the traditional and dominant technique for searching images for decades. However, not until recently have researchers started to realise some vital problems existing in CBIR systems. One of the most important is perhaps what people call the *semantic gap*, which refers to the gap between the information that can be extracted from images and the interpretation of the images for humans. As an attempt to bridge the semantic gap, automatic image annotation has been gaining more and more attentions in recent years.

This thesis aims to explore a number of different approaches to automatic image annotation and some related issues. It begins with an introduction into different techniques for image description, which forms the foundation of the research on image auto-annotation. The thesis then goes on to give an in-depth examination of some of the quality issues of the data-set used for evaluating auto-annotation systems. A series of approaches to auto-annotation are presented in the follow-up chapters. Firstly, we describe an approach that incorporates the salient based image representation into a statistical model for better annotation performance. Secondly, we explore the use of non-negative matrix factorisation (NMF), a matrix decomposition technique, for two tasks; object class detection and automatic annotation of images. The results imply that NMF is a promising sub-space technique for these purposes. Finally, we propose a model named the image based feature space (IBFS) model for linking image regions and keywords, and for image auto-annotation. Both image regions and keywords are mapped into the same space in which their relationships can be measured. The idea of multiple segmentations is then implemented in the model, and better results are achieved than using a single segmentation.

Contents

Acknowledgements	x
1 Introduction	1
1.1 Aims and objectives	2
1.2 Contributions	2
1.3 Thesis Structure	3
2 Background	5
2.1 Content Based Image Retrieval	5
2.1.1 Category Search	6
2.1.2 Target Search	6
2.1.3 Association Search	7
2.1.4 Image Search in the Real World	7
2.2 Automatic Image Annotation	8
2.2.1 Why Automatic Image Annotation?	9
2.2.1.1 Automatic Image Annotation vs. Manual Annotation . .	9
2.2.1.2 Automatic Image Annotation vs. Query-by-Example . .	10
2.2.2 Statistical Models	11
2.2.2.1 Co-occurrence Model	11
2.2.2.2 Machine Translation Model	11
2.2.2.3 Cross Media Relevance Model	12
2.2.2.4 Continuous Relevance Model	12
2.2.2.5 Other Probabilistic Approaches	13
2.2.3 Vector Space Related Approaches	14
2.2.3.1 The SvdCos Method	14
2.2.3.2 Saliency-based Semantic Propagation	14
2.2.3.3 Cross-Language Latent Semantic Indexing based Approach	15
2.2.4 Classification Approaches	16
2.2.4.1 Non-negative Matrix Factorization Approaches	16
2.2.4.2 Support Vector Machine Approaches	17
2.2.4.3 Multiple Instance Learning Approaches	17
2.2.5 Discussion	18
2.3 Evaluation of Annotation Effectiveness	19
2.3.1 Precision and Recall	19
2.3.1.1 Per-image Precision and Recall	20
2.3.1.2 Per-word Precision and Recall	20
2.3.2 Keyword Number with Recall>0	21

2.3.3	Normalized Score	21
2.4	Summary	21
3	Image Description	22
3.1	Region Choosing	22
3.1.1	Fixed Partitioning	23
3.1.2	Segmentation	23
3.1.3	Saliency	24
3.2	Feature Extraction	24
3.2.1	Colour	24
3.2.1.1	Colour Invariants	26
3.2.1.2	The MPEG-7 Colour Structure Descriptor	26
3.2.2	Shape	28
3.2.2.1	Moment of Inertia	29
3.2.2.2	The MPEG-7 Contour Shape Descriptor	29
3.2.3	Texture	31
3.2.3.1	Mean Oriented Energy	31
3.2.4	SIFT - a local descriptor for saliency	32
3.3	Feature Quantisation	32
3.3.1	The Self-Organizing Map (SOM)	33
3.3.1.1	The SOM Toolbox	33
3.3.1.2	Shape Clustering Using CSS and SOM	34
3.3.2	k -Means Clustering	36
3.4	Image Description Examples	36
3.4.1	The “blob” representation	36
3.4.2	Saliency based visual term representation	37
3.5	Summary	38
4	Quality Issues with Data-Sets	40
4.1	Two Image Collections	41
4.1.1	The Corel Set	41
4.1.2	The Yahoo Set	41
4.2	Three Auto-annotation Methods	44
4.2.1	The CSD-Prop Method	44
4.2.2	The SvdCos Method	44
4.2.3	The CSD-SVM Method	45
4.3	Evaluation Metrics	46
4.4	Results and Discussion	46
4.4.1	Comparison with state-of-the-art methods	46
4.4.2	An Examination of word combinations	47
4.4.3	Comparison between the three methods when different training sets are used.	48
4.5	Summary	49
5	Incorporating a Statistical Model with Salient Regions	53
5.1	Statistical Models for Image Annotation	54
5.1.1	The Cross-Media Relevance Model (CMRM)	54

5.2	Hybridising CMRM with a Saliency-based Image Representation	55
5.3	Results and Discussion	56
5.3.1	Experimental Results of Auto-annotation by Saliency-based CMRM	57
5.4	Summary	59
6	Non-negative Matrix Factorisation	63
6.1	Using Non-negative Matrix Factorization (NMF) to Discover Object Classes and Their Extent	63
6.1.1	NMF vs. PCA	64
6.1.1.1	Principal Component Analysis (PCA)	64
6.1.1.2	Non-negative Matrix Factorisation (NMF)	66
6.1.1.3	Relation of NMF and PCA	67
6.1.2	NMF for Object Class Detection	67
6.1.3	Experimental Results and Discussion	69
6.2	Auto-annotation via Semantic Propagation in Sub-space	71
6.2.1	NMF as an alternative to SVD	72
6.2.1.1	SVD	72
6.2.1.2	NMF for Sub-space Projection	73
	NMF with sparseness constraints	74
6.2.2	Using NMF and Semantic Propagation for Auto-annotation	75
6.2.3	Experimental results	76
6.2.3.1	The Washington Image Data-set	76
6.2.3.2	Performance Evaluation	76
6.2.3.3	Experiment Settings	76
6.2.3.4	Results	79
6.3	Summary	81
7	The Image Based Feature Space Model	84
7.1	An Image Based Feature Space and Mapping	85
7.1.1	Related Work	85
7.1.2	The Algorithm	86
7.1.2.1	Approach Overview	86
7.1.2.2	Representing image regions by salient regions	87
7.1.2.3	Image-Based Feature Mapping	87
7.1.2.4	Application to Region-Based Image Annotation	89
7.1.2.5	A Simple Example	89
7.1.3	Experimental Results and Discussion	91
7.1.3.1	Correspondence of segments and words	91
7.1.3.2	Results on Region Based Image Annotation	92
7.1.4	Summary	98
7.2	Multiple Segmentations for Image Auto-Annotation	98
7.2.1	The Algorithm	99
7.2.1.1	Approach Overview	99
7.2.1.2	Generating Multiple Segmentations	100
7.2.1.3	Incorporating Image Based Feature Space and Mapping with Multiple Segmentation	100
7.2.2	Experiment and Results	102

7.2.3	Summary	103
7.3	Summary	104
8	Conclusions and Future Work	106
8.1	Summary and Conclusions	106
8.1.1	Novel work of the thesis	109
8.2	Future Work	109
8.2.1	Image Description	109
8.2.2	Quality Issues with Data-Sets	110
8.2.3	Incorporating a Statistical Model with Salient Regions	111
8.2.4	Non-negative Matrix Factorisation for Image Auto-annotation . . .	111
8.2.5	The Image Based Feature Space Model	112
8.3	The Future of Automatic Image Annotation	113
	Bibliography	115

List of Figures

2.1	Google Image Search results of (a) “Washington”; (b) “Washington” with “face” constraints	8
3.1	Three steps of the process of image description.	22
3.2	Different forms of region choosing for image description.	25
3.3	Structuring elements for images with different size (only a part of the image is shown)	27
3.4	Example of contour evolution and corresponding CSS [Bober (2001)]	30
3.5	The original Camera-Man image and the result of being convoluted by 12 different oriented filter	31
3.6	Updating the best matching unit (BMU) and its neighbors towards the input sample marked with x. The solid and dashed lines correspond to situation before and after updating, respectively [Vesanto et al. (2000)]	34
3.7	Three random cells from the SOM of 1100 marine creature shapes	34
3.8	The SOM of 1100 marine creature shapes	35
3.9	The process of generating “blobs” for image description.	37
3.10	Image representation using quantised salient region descriptors (Hare, 2006). . .	39
4.1	Examples of similar Corel images, the number in the parenthesis being the file name of the image.	42
4.2	Examples of Yahoo images. The top images are inappropriately annotated, and for the bottom images only one object is annotated.	43
4.3	The curves show, on the Corel and Yahoo set respectively, the CSD Euclidean distance between each image and its nearest neighbour (NN).	43
4.4	(a) Four word combinations that are correctly predicted by CSD-Prop and CSD-SVM, being ordered by the number of occurrences of each combination in the Corel training set. (b) Four word combinations that are correctly predicted by CSD-Prop and CSD-SVM, and with the number of occurrences greater than 5 in the Coral training set.	50
4.5	Two combinations predicted by CSD-SVM that do not exist in the training set, words in bold being correct.	52
5.1	(a) Precision-Recall curves for several different auto-annotation methods. Error bars show range of precision over 100 repeated runs, each of which used a random separation of the Washington set into training, test and evaluation sets. (b) A zoomed in version of (a).	58
5.2	The Box-and-Whisker plot of the precisions of the saliency based CMRM approach (blue) and blob based CMRM approach (green). Results from 100 repeated runs are used. The horizontal axis represents the index of the predicted word, while the vertical axis represents the precisions. . . .	59

5.3	Per-word precision and recall of annotations predicted by Saliency-based CMRM and Region-based CMRM on the Washington set. (a) Per-word precision. (b) Per-word recall.	61
5.4	Example Annotations	62
6.1	Parts-based representation of faces learnt by NMF and holistic representation learnt by PCA (Lee and Seung, 1999).	65
6.2	Top segments for 6 (out of 35) object classes discovered in the LabelMe data-set. Note how the segments, learned from a collection of unlabeled images, correspond to trees (a), sky (b), buildings (c), leafless trees (d), roads (e). However, for the last group of segments (f), it is not obvious which class of objects it corresponds to. We consider it as the class of cars in our evaluations.	70
6.3	Graphical representation of SVD.	72
6.4	Illustration of the difference between NMF and SVD (Xu et al., 2003). . .	73
6.5	Illustrations of 4 different degrees of sparseness, 0.1, 0.4, 0.7, 0.9. The height of each bar denotes the value of one element of the the vector (Hoyer, 2004).	74
6.6	Sample images and their annotations from the Washington Ground Truth Image Database.	77
6.7	Plot of empirical keyword distribution in the Washington data-set	78
6.8	The normalised score (E_{ns}) of applying NMFsc for image auto-annotation. The closest training image ($M = 1$) is used for propagation. The degree of sparseness is varied from 0.5 to 0.9.	80
6.9	The normalised score (E_{ns}) of applying NMFsc for image auto-annotation. The top 2 closest training images ($M = 2$) is used for propagation. The degree of sparseness is varied from 0.5 to 0.9.	80
6.10	The normalised score (E_{ns}) of applying NMFsc for image auto-annotation. The top 3 closest training images ($M = 3$) is used for propagation. The degree of sparseness is varied from 0.5 to 0.9.	81
6.11	The normalised score (E_{ns}) of applying CNMF for image auto-annotation. The curve “CNMF 1” represents the results when $M = 1$. “CNMF 2” and “CNMF 3” represent the case when $M = 2$ and $M = 3$	81
6.12	Precision-Recall curves for several different semantic propagation-based image auto-annotation methods. Error bars show range of precision over 100 repeated runs, each of which used a random separation of the Washington set into training and test sets.	82
6.13	Some sample results of image auto-annotation using the classic NMF (CNMF) and NMF with sparseness constraints (NMFsc).	82
7.1	Examples of globally annotated images from the Washington data-set (University of Washington, 2004).	85
7.2	Examples of locally annotated images from the LabelMe data-set (Russell et al., 2006).	85
7.3	Diagram of a simple example of using the image based feature space for relating image regions and words	90
7.4	Top 25 Words that appear most frequently in the Washington set	91
7.5	Segmentation samples of images from the Washington data-set (University of Washington, 2004).	93

7.6	The number of times words “Track”, “Stand” and “Football Field” occur together and separately.	94
7.7	Samples of colour images containing “Flower” and the counterpart gray ones from the Washington data-set (University of Washington, 2004). The top ones are RGB colour images, and the bottom ones are the corresponding gray images.	94
7.8	Some good results of representative regions found by the image based feature space approach for the corresponding words.	95
7.9	Some bad results of representative regions found by the image based feature space approach for the corresponding words.	97
7.10	Some good examples of image region annotation through the image based feature space approach.	97
7.11	Some bad examples of image region annotation through the image based feature space approach.	98
7.12	Examples of segmented images at different levels of segmentation	100
7.13	The number of keywords with recall>0 for each approach at different values of threshold	104
7.14	The total number of correctly predicted words for each approach at different values of threshold	104
7.15	Some annotation examples by multiple segmentation based approach . .	105

List of Tables

3.1	HMMD Colour Space Quantization for CSD	28
3.2	Components of Curvature Scale-Space Descriptor [Manjunath et al. (2002)]	30
4.1	Illustration of propagation for the CSD-Prop method	44
4.2	Comparison between CSD-Prop, SvdCos, CSD-SVD and some other state-of-the-art methods using the Corel images	51
4.3	Comparison between the three methods on different training sets	51
5.1	Summary of Results	60
6.1	Segmentation accuracy (ρ) of the top 20 segments returned by NMF on four object classes from the LabelMe dataset. It is compared with the results from Russell et al. (2006) on the same data-set.	71
6.2	Summary of results of image auto-annotation using several different semantic propagation-based methods.	83
7.1	The number of correct segments out of the top 25 for our method and random choice.	96
7.2	Performance comparison of Machine Translation Model (MT), Point-wise Diverse Density Model (PWDD) and Image Based Feature Space Model (IBFS).	98
7.3	Region features	102
7.4	Performance comparison of using multiple segmentations for image auto-annotation with single segmentation	103

Acknowledgements

Firstly, I would like to thank Paul Lewis for his supervision. His wise academic advice and ideas have played an extremely important role in the work presented in this thesis. Without Paul's support, this thesis would not have been possible.

Secondly, I would like to thank my friends and colleagues in the IAM group - in particular, Jonathon Hare, Patrick Sinclair, Simon Goodall, Wasara Rodhetbhai, Sebastian Stein and Jaime Cerda Jacobo - for their inspirations and discussions. I would also like to thank Tao Guan, Leran Wang, Liwei Hao, Jinyan Zhou, Yaozhong Liang and Xutao Kuang for their help in my daily life.

Finally, I would like to thank my parents for their everlasting love and support.

Chapter 1

Introduction

In the last decade, digital imagery has grown at a phenomenal speed in many directions, resulting in an explosion in the number and size of image archives required to be organized. In particular, with the widespread use of digital cameras, mobile phones with built-in cameras, and storage of personal computers reaching to a level of hundreds of gigabytes, individuals nowadays can easily produce thousands of personal images. Meanwhile, photo sharing through the Internet is becoming more and more popular. For example, by June 2005, the Internet photo-sharing website Flickr¹ had almost one million registered users and hosted 19.5 million photos, with a growth of about 30 percent per month (Li and Wang, 2006). By April 2007, Flickr had over 5 million registered users and over 250 million images (Ames and Naaman, 2007). Although people like the Flickr users are increasingly attaching tags to their images, the vast majority of the images on the Internet are barely documented, making it very difficult for people to find one of interest. In order to handle overload and exploit the massive image information, we need to develop techniques to document and search images.

Earlier efforts in image search has been mainly focused on content-based image retrieval (CBIR), which retrieves images by analysing and comparing low-level image information. CBIR systems usually rely on the feeding of desired image examples from the user as a starting point, and is known as the query by example paradigm. However, unskilled users may find it tedious to do so. Query by semantics (e.g. textual words) is more preferred. For example, if you want to find sunset images, it is more convenient and explicit to directly use the semantic word “sunset” as the query, than to find a sunset image or draw a sunset-like figure first. Intuitively, image retrieval would be more straightforward if all the images in the database were semantically annotated. By using standard text query techniques, images could be found in a manner that would meet the different needs of many users. Moreover, by combining these text-based approaches with visual content search techniques, users could have much more control over the search. In this thesis,

¹www.flickr.com

we attempt to investigate the feasibility and issues of automatically annotating images with textual words through analysis of the image contents. The terms ‘annotations’, ‘captions’ and ‘labels’ in this work all refer to textual keywords that describe the content of a particular image.

1.1 Aims and objectives

The original aims of this work were to investigate promising approaches to automatic image annotation, by either utilising and modifying different techniques in the information retrieval community, especially content-based image retrieval, or developing new techniques. The efforts toward achieving this objective consist of two intertwined parts; the development of a suitable image description, and the development of an advanced machine learning technique for associating words with images. In this work, we will explore both areas, though the second is the focus of the research.

In the process of pursuing the objective, the work has also investigated some relevant issues and tasks regarding automatic image annotation. For example, we have closely examined some quality issues of the data-sets used for experiments on image auto-annotation. We have also made a step forward to attempt to relate words with specific regions of images, which is considered as a task of object detection.

1.2 Contributions

This thesis brings a number of contributions to the field of automatic image annotation. They are itemised briefly as follows.

- An in depth investigation into some of the quality issues with image data-sets that are used for research on image auto-annotation.
- The development of an approach to image auto-annotation using salient region description of images with a statistical model.
- The development of an approach to finding object classes in an unlabelled image collection, using the non-negative matrix factorisation (NMF) technique.
- The demonstration of the potentials of NMF as an alternative approach to latent semantic indexing (LSI) for automatic image annotation through semantic propagation.
- The development of a model named the image based feature space (IBFS) model for linking image regions or segments with text labels, as well as for automatic image annotation.

- The development of an approach to improving annotation performance using multiple segmentations of images.

The research has led to one refereed journal publication, four refereed conference papers and one refereed workshop paper. Tang and Lewis (2006) demonstrated some problems of experimentations on automatic image annotation using the popular Corel set compared with the Yahoo based training set. Tang and Lewis (2007a) gave more extensive and detailed experiments and discussions on this subject. Tang et al. (2006) proposed the novel use of the salient region description of images with a statistical model. Tang and Lewis (2008) explored the use of non-negative matrix factorisation (NMF) for object class detection and auto-annotation of images. Tang and Lewis (2007b) invented a model named image based feature space model for discovering the relations between image regions and text labels. This model is also extended to image auto-annotation. Tang and Lewis (2007c) proposed to incorporate multiple levels of segmentation of images for auto-annotation, in order to improve the results.

1.3 Thesis Structure

This thesis presents the work carried out by the author in attempting to achieve the goals outlined earlier in Section 1.1. The structure and content of the thesis is described in the following on a chapter by chapter basis.

- **Chapter 2 - Background** Introduction the background behind this work. Research on traditional content-based image retrieval is introduced, in the form of three different retrieval scenarios. It then goes on to review a number of very different automatic image annotation techniques in the literature. Finally it describes several performance evaluation metrics used in this work.
- **Chapter 3 - Image Description** Presents a variety of techniques regarding the description of the information encapsulated in images. It describes the process of image description as three steps - region choosing, feature extraction and feature quantisation. The chapter ends with the introduction of two description examples that are used in the research undertaken by the author.
- **Chapter 4 - Quality Issues with Data-Sets** Examines some quality issues of the widely adopted data-set, the Coral set, for evaluation. The analysis is undertaken by comparing the results of applying three different auto-annotation techniques on the Coral set and Yahoo set, which is constructed by the author.
- **Chapter 5 - Incorporating a Statistical Model with Salient Regions** Proposes a novel auto-annotation technique that incorporates the salient region representation of images with a statistical model.

- **Chapter 6 - Non-negative Matrix Factorisation** Explores the use of the non-negative matrix factorisation technique for automatic image annotation and object detection. The first part utilises NMF as a technique for discovering the object classes from a collection of un-annotated images. The second part demonstrates the potentials of NMF as an alternative approach to latent semantic indexing for image auto-annotation.
- **Chapter 7 - The Image Based Feature Space Model** Describes a model developed by the author, namely the image based feature space model. This model is demonstrated to be capable of relating keywords with image regions within the same feature space. It is then used to annotate images at both the local and global image level. The second part of this chapter incorporates the idea of multiple segmentations into the model and shows that better performance can be achieved.
- **Chapter 8 - Conclusions and Future Work** Discusses and concludes the overall results and contributions from the work presented in previous chapters. The chapter ends with some pointers to the future work regarding chapter 4 to 7, and an overview of the future of automatic image annotation from the author's point of view.

Chapter 2

Background

The aim of this chapter is to present an overview of the research that is related to this thesis, with focus on automatic image annotation techniques. Firstly, we briefly review the field of content based image retrieval, which serves as the foundation of image auto-annotation. Secondly, we discuss a number of auto-annotation techniques found in the literature, which are categorized into three main groups. Finally, several evaluation metrics for performance comparison on auto-annotation are introduced.

2.1 Content Based Image Retrieval

Image retrieval has been an active research area since the 1970's (Rui et al., 1999). Researchers from two different communities, database management and computer vision, proposed two different directions of retrieval, one being text-based and the other visual based. Text-based image retrieval in the past usually required the images to be annotated with words manually before retrieval can be conducted. With the significant advances of database management and textual information retrieval, text-based image retrieval in this framework has achieved some success. However, two major difficulties make this approach unfavorable, especially when a large number of images are involved. The first one is simply the vast amount of labor needed for manual annotation. The second is due to the subjectivity of the annotators; individual persons may perceive images in very different ways, resulting in different labels.

In the early 1990's, Content-Based Image Retrieval (CBIR) emerged as a new technique and started to gain more and more attention. CBIR retrieves images based on the visual content, such as colours and textures, rather than the keywords. Smeulders et al. (2000) gave a thorough overview of CBIR techniques in the literature by the year 2000. Some more recent work on CBIR and also automatic image annotation can be found in the review by Datta et al. (2005). According to the categorization of Cox et al. (2000),

image retrieval tasks fall into three different scenarios: category search, target search and association search. In the following, we describe each category in details, together with a discussion of some important relevant work.

2.1.1 Category Search

In category search, the users search for one or more arbitrary image representatives that belong to a prototypical category, such as “cars”, “pedestrians” or “human faces”. Since each category is generally an object class, category search can also be considered as object class recognition.

Early research on object class recognition was mainly focused on face detection (Viola and Jones, 2001), where detection algorithms try to locate human faces in images if there are any, i.e. which image patch is a face. Afterwards, the research area has been extended to generic object detection. For example, Fergus et al. (2003) proposed an “unsupervised scale-invariant learning” approach to discovering object classes including “motorbikes”, “faces”, “airplanes” and “spotted cats” from a data-set containing background images. Objects were modeled as constellations of parts (i.e. salient regions), which were found by a salient region detection algorithm. For each object class, a probabilistic model was estimated on the training set to describe the relationships between the scale, location and appearance of parts. Sivic et al. (2005) and Russell et al. (2006) described the aims of their experiments as: “Is it possible to learn visual object classes simply from looking at images?”. In other words, they took away the labels from the training set and tried to find object classes purely on the visual information of the images. Unsupervised topic discovery techniques, probabilistic Latent Semantic Analysis (pLSA) (Hofmann, 1999, 2001) and Latent Dirichlet Allocation (LDA) (Blei et al., 2003), were borrowed from the text analysis community. Images were treated as text documents and quantised salient feature descriptors found in images were treated as words. Therefore, discovering object classes from images is analogous to discovering topics from a corpus of text documents.

2.1.2 Target Search

In target search, the users have a precise copy of the image in mind and intended to find the exact same or similar ones from the database. Applications where this type of search is useful include checking if a particular trademark logo has been registered (Eakins et al., 1998), or finding relevant information (e.g. title, artist, etc.) of a specific painting (Hare and Lewis, 2005a). Target search usually resorts to a sample image given by the user, and is known as the “query by example” paradigm. Therefore, rather than asking the system to “find images with trees” as in the category search scenario, here the users ask it to “find similar images to this one”. Sample images can be images the user already possessed or sketches drawn by the user.

Jacobs et al. (1995) proposed an approach to retrieving images that are similar to a hand-drawn sketch submitted by the user. The similarity between the sketch and each image from the database is calculated as the distance of the coefficients generated by wavelet decomposition of the images. However, in this retrieval mechanism, information about the desired target image can only be expressed to a very limited extent, considering the quality of the sketch. A more general choice is using example images, i.e. an image containing the object or content to be searched for. Numerous researchers have adopted this mechanism in their experiments (Flickner et al., 1995; Mokhtarian et al., 1996; Wang et al., 2001; Hare and Lewis, 2004). Unfortunately, because of the phenomenon which is now commonly referred to as the semantic gap (Smeulders et al., 2000), this mechanism often fails to capture the similarity that is latent but can be inferred by humans. Besides, finding a suitable example image is a difficult task in itself (Rodden, 1999).

2.1.3 Association Search

Association search is also known as browsing, where users have no particular target image in mind at the beginning of the search. Often, the goal of the search may change, and the users may refine the search through interaction with the system in an iterative manner. Interaction is usually achieved by relevance feedback (Rui et al., 1997; Hiroike et al., 1999) from the user.

A common choice of organising data for browsing is hierarchical structures, which is also known as tree structures. For example, Barnard and Forsyth (2001) proposed a generative hierarchical model for organising images. Both labels and image segments were integrated into the construction of the model. Because of the hierarchical characteristic of the model, image browsing is well supported. An alternative structure is image networks. More flexible navigation is supported in networks, because they need not be acyclic as trees.

2.1.4 Image Search in the Real World

Google Image Search¹ is probably the most famous real world search engine for images. There are also many similar engines such as Yahoo Image Search², and MSN Image Search³, and so on. As many researchers have pointed out, these search engines rely on textual descriptions found on the Web pages containing the images and the file names of the images (Li and Wang, 2006). Although the results of search are fairly good in many cases (e.g. searching for images of a celebrity), one should be aware that the

¹<http://images.google.com>

²<http://images.search.yahoo.com>

³<http://images.live.com>



FIGURE 2.1: Google Image Search results of (a) “Washington”; (b) “Washington” with “face” constraints

textual descriptions are mostly given by people manually. Considering the effort of manual annotation, the number of images with textual descriptions is probably very small compared with those with no descriptions. In other words, the results are chosen from candidates that constitute only a very small portion of the images actually available on the Internet.

With the advances of research in CBIR, there have been already some realisations of CBIR techniques in real world applications. For example, Google started to integrate face detection techniques into their image search engine. Although the new feature has not been released yet, users can use it by appending “&imgtype=face” to the end of the URL. Figure 2.1 shows the result of a search for images using keyword “Washington” and that of applying “face” constraints. As can be seen, results for “Washington” contains images of maps, places and persons, while that for “Washington” with constraints of “face” filtered out all non-face images. The technique proposed by Jacobs et al. (1995) has been efficiently implemented in Retrievr⁴, a sketch based image search interface for retrieving images from the online photo sharing website Flickr.

2.2 Automatic Image Annotation

In the past, research on content-based image retrieval tended to focus on the technique side, ignoring the user side. Despite this, some researchers have examined the design of retrieval systems and provided some insights (Smeulders et al., 2000; Hollink et al., 2004). In particular, the problem of what is now called the *semantic gap* has been highlighted. Smeulders et al. (2000) described semantic gap as follows.

⁴<http://labs.systemone.at/retrievr/>

“The semantic gap is the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation.”

They also concluded that:

“A critical point in the advancement of content-based retrieval is the semantic gap, where the meaning of an image is rarely self-evident. The aim of content-based retrieval systems must be to provide maximum support in bridging the semantic gap between the simplicity of available visual features and the richness of the user semantics.”

As an attempt to bridge the semantic gap, automatic image annotation techniques have attracted a lot of interest in recent years. The aim of auto-annotation techniques is to attach textual labels to un-annotated images, as the descriptions of the content or objects in the images. In this thesis, we are interested in automatic annotation of public domain images (see Figure 6.6 for examples), as different from more specific domain images, such as medical images (Deselaers et al., 2007). In the following, we will first discuss the advantages of automatic image annotation. Then, we will review a number of auto-annotation techniques that have been published in the literature. The chosen techniques have been grouped into three categories: statistical models, vector space related approaches and classification approaches.

2.2.1 Why Automatic Image Annotation?

The final goal of automatic image annotation is mostly to assist image retrieval by supplying users with a text based interface for search. If successful, images can be retrieved in a way that is similar to search of text documents as many people do on Google. We discuss the reasons for automatic image annotation from two perspectives, manual annotation and CBIR.

2.2.1.1 Automatic Image Annotation vs. Manual Annotation

One of the most cited arguments supporting the use of automatic image annotation is probably that manual annotation requires enormous human effort. As we have said, countless images exist in our lives. It is not possible to annotate them all by hand. Automatic annotation by computer is a potential and promising solution to this problem.

However, a few issues exist with auto-annotation of images. For example, Enser et al. (2005) pointed out two limitations of automatic image annotation as compared with

manual annotation by experts. The first limitation, visibility limitation, is that keywords predicted by auto-annotation techniques usually have to be related to visible entities within the image, but users are frequently interested in the *significance* of objects or scenes displayed. In other words, conceptual content and contextual information which do not come with any visually salient features are difficult to capture. The second limitation, generic object limitation, refers to the generic nature of labels in the vocabularies. As Enser et al. (2005) described, “they have the common property of visual stimuli which require a minimally-interpretive response from the viewer”. They pointed out that search requests for images with features uniquely identified by proper name, such as identification, are very common. Again, visual saliency can not give much help.

We agree with the above arguments on limitations, but we also believe in the value of image auto-annotation. Although it is not possible (at least for the time being) to annotate conceptual interpretation of images, being able to automatically annotate simple objects depicted in images would be of great help, considering the quantity of images nowadays.

2.2.1.2 Automatic Image Annotation vs. Query-by-Example

Generally, query-by-example CBIR systems answer users’ search requests by calculating the visual similarities between the example and images in the database. This mechanism is useful when visual appearances of the images are more important, rather than the objects or concepts depicted. In particular, if the image contents do not have intuitive meanings, such as textile images and trademark images (Eakins et al., 1998), query-by-example seems to be the best choice. However, some limitations make auto-annotation a better alternative.

- *Unfavorable interface.* Query-by-example retrieval requires the user to submit a sample image in order for the system to perform a search. Users may find it difficult to obtain such examples. Although drawing a sketch is a potential way to get around this, users may feel it tedious to do so frequently. After all, the most straightforward way to submit a request is perhaps through a single word or two, indicating the desired content. This procedure matches well with the goal of automatic image annotation.
- *Computational cost.* Query-by-example requires similarity computations between the query image and a possibly large number of images in the database, which is time-consuming. Designing a system that supplies real-time responses could be a challenge. Automatic image annotation, on the other hand, is generally performed offline. Therefore, it is easier to achieve fast responses at the time of query.

2.2.2 Statistical Models

Statistical techniques have been popular in the field of information retrieval. Recently, researchers began to apply them to image auto-annotation, with very promising results. The basic idea of statistical techniques is to estimate the probabilities of documents related to the query submitted by the user. Documents are then ranked according to their probabilities. In the following, we review a number of statistical models which have been proposed and applied to image annotation in the recent years.

2.2.2.1 Co-occurrence Model

The co-occurrence model proposed by Mori et al. (1999) is perhaps one of the first attempts at image auto-annotation. They first divide images into rectangular tiles of the same size, and calculate a feature descriptor of colour and texture for each tile. All the descriptors are clustered into a number of groups, each of which is represented by the centroid. On the other hand, each tile inherits the whole set of labels from the original image. Then, they estimate the probability of a label w related to a cluster c by the co-occurrence of the label and the image tiles within the cluster, as follows

$$p(w|c) = \frac{m_{c,w}}{\sum_w m_{c,w}}$$

where $m_{c,w}$ is the number of times word w occurs with an image tile from cluster c . Given an un-annotated image q , they divide it into rectangular tiles and extract feature descriptors in the same way as before, and then find the closest cluster centroid for each tile. At this point, the probability of each label related to each of the test tiles can be measured. The labels having the highest average probabilities over all the tiles of a test image are chosen as the predictions. Mathematically, it can be denoted as follows

$$p(w|q) = \frac{1}{|q|} \sum_{t \in q} p(w|c_t)$$

where $p(w|q)$ is the average probability of w given image q , c_t is the closest cluster of tile t from q , and $|q|$ is the number of tiles.

2.2.2.2 Machine Translation Model

Duygulu et al. (2002) proposed a machine translation model for image auto-annotation. They argued that region based image annotation is more interesting because global annotation does not give information on which part of the image is related to which label. In their point of view, the process of attaching labels to image regions is analogous to the translation of one form of representation (image regions; French) to another form (labels; English).

They first use a segmentation algorithm to segment images into object-shaped regions. Then, feature quantisation is applied to the feature descriptors that are extracted from all the regions, to build a visual vocabulary called ‘blobs’. A ‘blob’ is in fact a representative of a cluster of visually similar image regions. Finally, a machine translation model which was initially proposed for linguistic translation is adopted to build a ‘lexicon’, a translation table containing the probability estimations of the translation between image regions and labels. An unseen image is annotated by choosing the most likely word for each of its regions.

2.2.2.3 Cross Media Relevance Model

Jeon et al. (2003) improved on the results of Duygulu et al. (2002) by introducing a generative language model to image annotation, referred to as the cross-media relevance model (CMRM). The same process as used by Duygulu et al. (2002) was chosen to calculate the blob representation of images. However, as different from the assumption by Duygulu et al. that there exists an underlying one-to-one correspondence between the blobs and words, they only assume that a set of blobs is related to a set of words. Thus, instead of seeking a probabilistic translation table, CMRM simply approximates the probability of observing a set of blobs and words in a given image.

It is assumed that for a given un-annotated image I , there exists an underlying probability distribution (denoted as $P(\cdot|I)$) of all possible blobs and words that could appear in image I . If the blob representation of I is $I = \{b_1 \dots b_m\}$, where m is the number of blobs in I , the probability of observing word w is approximated as

$$P(w|I) \approx P(w|b_1, \dots, b_m)$$

Once the image is chosen, calculating $P(w|b_1, \dots, b_m)$ is equivalent to calculating the joint probability $P(w, b_1, \dots, b_m)$, which is approximated as the expectation over the entire training set. Under the assumption that words and blobs are generated independently given a training image J , $P(w, b_1, \dots, b_m)$ is calculated as follows

$$P(w, b_1, \dots, b_n) = \sum_{J \in T} P(J) P(w|J) \prod_{i=1}^m P(b_i|J)$$

where T is the training set. Prior probabilities $P(J)$ are kept uniform over all training images, while $P(w|J)$ and $P(b_i|J)$ are estimated by smoothed maximum likelihood.

2.2.2.4 Continuous Relevance Model

Lavrenko et al. (2003) argued that the process of quantization from continuous image features into discrete blobs, as the approach used by the machine translation model

and the CMRM model, will cause the loss of useful information in image regions. By using continuous probability density functions to estimate the probability of observing a region given an image, they improved on the results obtained by Duygulu et al. (2002); Jeon et al. (2003).

Specifically, they replace $P(b_i|J)$ of the CMRM model with $P(v_i|J)$, where v_i is the continuous feature vector of an image region. $P(v|J)$ is a non-parametric kernel-based density estimate as follows

$$P(v|J) = \frac{1}{n} \sum_{i=1}^n \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} e^{-(v-v_i)^\top \Sigma^{-1} (v-v_i)}$$

where n is the number of region feature vectors in image J , and Σ is the covariance matrix for controlling the degree of smoothing. In their work, Σ is simply set to $\Sigma = \beta \cdot I$, where I is the identity matrix. β is a value selected empirically on a held-out portion of the training set. Feng et al. (2004) modified the above model such that the probability of observing labels given an image ($P(w|J)$) is modeled as a multiple-Bernoulli distribution. In addition, they simply divided images into rectangular tiles instead of applying automatic segmentation algorithms. Their Multiple Bernoulli Relevance Model (MBRM) achieved further improvement on performance.

2.2.2.5 Other Probabilistic Approaches

Monay and Gatica-Perez (2003) applied probabilistic latent semantic analysis (PLSA) (Hofmann, 1999) to image auto-annotation. Both text words and image features were treated as terms. It was assumed that each term can come from a number of the latent topics, and each image can contain multiple topics. However, in their experiments, the result of PLSA was worse than that of latent semantic indexing (LSI) (Deerwester et al., 1990) based on propagation. Blei and Jordan (2003) described three probabilistic models which are built upon the assumption that images and words are generated by a mixture of latent factors, each model corresponding to the way images and words are generated. The Gaussian-multinomial mixture model assumes that the entire image and captions are conditional on the same factor, while the Gaussian-multinomial LDA model assumes the image regions and captions are conditional on two disparate sets of factors. However, both are claimed to have some limitations. The third model, correspondence LDA, is a compromise of the former two, assuming that the image regions can be conditional on any factors, but captions can only be conditional on factors that already exist in the images. Experiments showed that the third model outperforms the other two. Carbonetto et al. (2004) proposed a model that takes into account the spatial relationships between objects or regions in images; their results showed that spatial context improves the accuracy of object recognition. Yavlinsky et al. (2005) showed that their non-parametric models achieve comparable results with state-of-the-art methods (Lavrenko et al., 2003; Feng

et al., 2004), even by using very simple global image features like colour and texture. Carneiro et al. (2007) proposed to estimate the semantic class distributions through a “pooling” process that is justified by Multiple Instance Learning (MIL) (Maron and Lozano-Pérez, 1998), without the need to segment images.

2.2.3 Vector Space Related Approaches

The vector space model framework is another popular technique in information retrieval, especially text retrieval. Generally, documents are represented as vectors, each of which contains the occurrences of words within the document in question. The length of the vectors is equal to the vocabulary size. In this section, several image auto-annotation approaches that utilize the vector space model are introduced. These approaches treat images as documents, and build visual terms which are analogous to words, from the image feature descriptors.

2.2.3.1 The SvdCos Method

Pan et al. (2004) proposed a series of auto-annotation methods which capture the association between words and blobs (Duygulu et al., 2002) through their pattern of occurrence over the entire training set. According to their reported results, the SvdCos method achieved the best performance. It works as follows.

Firstly, they constructed a N -by- $(W+B)$ data matrix $D = [D_W | D_B]$, where $D_W(i, j)$ is the count of label w_j in image I_i , and $D_B(i, j)$ is the count of blob b_j in image I_i . After weighting the matrix D according to the uniqueness of every kind of blobs and words, they applied singular value decomposition (SVD) in order to “clean up noise and reveal informative structure”. The largest singular values that preserve 90% of the variance were kept and others were set to zero. The matrix after SVD is denoted as $D_{svd} = [D_{W,svd} | D_{B,svd}]$. Then, they calculated a translation table T , where $T_{i,j}$ is the cosine value of the angle between the i th column vector of D_W and j th column vector of D_B , i.e. $T_{i,j} = \cos(D_W(i), D_B(j))$. Given a query image with a blob representation as $q = [q_1, \dots, q_B]$, the words to be predicted can be chosen from the term-likelihood vector $p = Tq$, where $p = [p_1, \dots, p_W]^T$, p_i being the likelihood of label w_i .

2.2.3.2 Saliency-based Semantic Propagation

Hare and Lewis (2005c) proposed a model of automatic image annotation via propagation of words. Their work is based on the premise that visually similar images should have similar semantic content. For a given un-annotated image, they ranked all the training images and chose the labels directly from the top images.

The method they proposed is based on the concept of Latent Semantic Indexing (LSI). Images are projected into a sub-space in order to reveal the underlying semantic structure of the data-set. Firstly, salient regions were chosen from the peaks in a multi-scale difference-of-Gaussian (DoG) pyramid (Lowe, 1999) of each image. Each salient region was then described by a Scale Invariant Feature Transform (SIFT) descriptor (Lowe, 1999). They quantised the whole set of SIFT descriptors from the training set into 3000 classes using k-means clustering algorithm. Each class is considered as a visual term. As a result, each image can be represented by a histogram of visual terms. They built a term-by-document matrix A where A_{ij} indicates the number of occurrence of the i th visual term in the j th image. SVD was utilized to decompose A into a sub-space, resulting in the product of three matrices U, Σ, V , $A = U\Sigma V^T$. The k largest singular values of Σ were chosen to generate $A_k = U_k \Sigma_k V_k^T$. A_k is an approximation of A and is expected to be noise free. Given a query vector q , it can be also projected to the sub-space built above as $q_{sub} = q^T U_k \Sigma_k^{-1}$. The similarity between q and each of the training images is measured as the cosine value of the angle of q_{sub} and each column of V^T . Labels of the top ranked images (1, 2 and 3 in their experiments) were propagated to q .

Hare and Lewis (2005c) also developed a simple vector space model which directly compares the visual term histograms of images and applies propagation based on the similarities of the histograms. They reported that the LSI based method achieved better results than the vector space method on the same data-set.

2.2.3.3 Cross-Language Latent Semantic Indexing based Approach

Dumais et al. (1997) have demonstrated that Latent Semantic Indexing (LSI) can be used for cross-language information retrieval. Their system can perform text searching on a collection of French and English documents where queries could be in either language. This was realized by applying SVD to the term-by-document matrix in which the term vectors contain both French and English terms. As a result, the documents are projected into a low dimensional sub-space where co-occurrences of words from different languages were captured. Documents that are only in one language can then be mapped into the space and queried by keywords from the other language.

Hare et al. (2006) extended this approach to image retrieval of un-annotated images through keyword queries, without actually annotating the images. They consider the visual terms and text labels of each image as two documents in two forms of terms (or “languages”), one being the translation of the other. First, they build a training term-by-document matrix O , each term vector of which consists of both visual terms and text labels. Matrix factorization is applied to turn O into two separate matrices as

$$O = TD$$

where T is called the term matrix and D is the document matrix. Then, they build an partially observed matrix for the test images, with any un-observed terms (text labels in their case) set to zeros. The document-space of the test matrix is created using the term matrix (T) from the training matrix as a basis. As a result, all the un-observed terms in the test matrix are given pseudo-values, which are used for keyword based retrieval. In terms of auto-annotation, these values also indicate the likelihood of a label related to an image.

2.2.4 Classification Approaches

Classification approaches for image auto-annotation view the process of attaching words to images as that of classifying images to a number of pre-defined groups, each of which is characterised by a concept or word. Given an unannotated image, the classification algorithms find its membership and annotate it with the corresponding word. Multiple annotations can be generated by assuming an image belongs to multiple classes.

2.2.4.1 Non-negative Matrix Factorization Approaches

Non-negative matrix factorization (NMF) (Lee and Seung, 1999) is a matrix factorization technique that has become popular recently. Because of its non-negative constraints, many researchers (Tsuge et al., 2001; Guillaumet et al., 2002; Xu et al., 2003; Liu and Zheng, 2004) from the information retrieval community regard it as more suitable for part-based representation of data, such as text documents and images, and for further applications such as classification or retrieval.

Tsuge et al. (2001) used NMF as a sub-space technique to project text documents into a low dimensional semantic space, where the distances between query documents and those in the database are measured. Xu et al. (2003) adopted NMF for document classification. They factor the term-by-document matrix X into a basis matrix U and coefficient matrix V . The membership of a document is chosen as the one with the maximum value in the corresponding column of V .

Guillaumet et al. (2002, 2003) used NMF for image classification. They build a collection of image patches which were categorized into 10 classes. Both the training set and test set are built by randomly choosing 1000 patches respectively. For the training patches from each of the 10 classes, they apply NMF in order to map them into a sub-space, in which a classifier is learned afterwards. Given a test image to classify, they project it to all the 10 sub-spaces built from the training set and choose the one which achieves the high value based on the classifiers. Guillaumet and Vitrià (2003) compared several different distance metrics for the space defined by the bases discovered via NMF. Interestingly they found that in their experiments on object classification, when occlusions are present NMF with

the cosine distance measure performs the best in comparison with PCA approaches. Liu and Zheng (2004) argued that the bases learned by NMF are not directly suitable for object recognition using nearest neighbour methods. They proposed to orthonormalize the bases before further analysis, and demonstrated that object recognition accuracy can be improved in this way.

2.2.4.2 Support Vector Machine Approaches

The Support Vector Machine (SVM) is a popular and powerful learning technique for data classification. Given a set of data points which belong to one or two classes, a linear SVM finds a hyperplane that leaves the largest number of points from the same class on the same side, while maximizing the distances of both classes to the hyperplane. Because of its high generalization performance, it has been introduced to the image community in which features are usually of very high dimensionality.

Chapelle et al. (1999) were one of the earliest to apply SVM to image classification. Basic colour histograms are extracted for image description. Since SVMs are initially designed for binary classification, in their experiments on multiple class (seven classes of Coral images) classification, a “one against the others” SVM algorithm is used. Recently, Gao et al. (2006) proposed a framework for image region classification using multiple SVM classifiers. Firstly, a multi-resolution grid-based representation of images is generated, where images are divided into regular tiles at different levels. From each tile, a 90-dimensional multi-modal visual feature descriptor is extracted. The 90-dimensional heterogeneous feature space is partitioned into 9 single-modal homogeneous subsets. Secondly, a weak SVM classifier is learned for each feature subset given an object class or concept. The most effective classifiers and the corresponding feature subsets and tile sizes are selected. Finally, the chosen weak SVM classifiers are combined to form an optimal classifier or ensemble classifier using the Boosting technique. By merging neighboring image tiles which are classified into the same class, their approach is able to annotate images at the object level. Qi and Han (2007) combined two sets of SVMs for image auto-annotation. One set is fed with regional image features which are found by the Multiple Instance Learning (MIL) (Maron and Lozano-Pérez, 1998) technique, while the other uses global image features. The outputs of both sets of SVMs are incorporated to annotate test images.

2.2.4.3 Multiple Instance Learning Approaches

As termed by Maron and Lozano-Pérez (1998), “Multiple-instance learning (MIL) is a variation on supervised learning, where the task is to learn a concept given positive and negative bags of instances”. A typical MIL problem can be described as follows. A “bag” can be considered as a container which contains a number of “instances”. An “instance”

is an object or observation which can be categorized into one of the two classes, positive or negative. However, instead of labeling each individual instance, only the bags are labeled. If all the instances from a bag are negative, then the bag is labeled as negative; if one or more instances are positive, then the bag is labeled as positive. The purpose of MIL is to learn a concept which is able to correctly label individual instances. Maron and Lozano-Pérez (1998) proposed a framework called Diverse Density (DD) to solve the MIL problem. The DD value of a point measures “how many different positive bags have instances near that point, and how far the negative instances are from the point”.

Some researchers consider an image as a bag of instances, each corresponding to a region within the image. Given a keyword, images annotated with it are positive bags, while the others are negative ones. A number of approaches that use MIL techniques for image categorization and auto-annotation have been proposed. Chen and Wang (2004) proposed their DD-SVM framework for image categorization. DD-SVM is a MIL method that incorporates the Diverse Density framework with the support vector machines (SVM) technique. Firstly, they segment images into regions, and extract a 9-dimensional feature descriptor from each of them as an instance. Secondly, a collection of instance prototypes are learnt according to a DD function. Each instance prototype is the representative of a class of instances that are more likely to appear in bags with a particular label than in the others. All the prototypes are used to build a space in which each axis corresponds to a prototype. Then, training bags (images) are mapped into the space. The coordinate of a bag on a particular axis is the distance of the corresponding prototype to the closest instance from the bag. Finally, A standard SVM is trained based on the positions of the training bags in the space and the corresponding labels. Test images can be categorized by SVM based on their positions in the above built space. Bi et al. (2005) used a similar approach, except that instead of using the DD framework, sparse SVM is adopted for prototype selection and classifier construction. It is reported as more efficient. Yang et al. (2005) apply a modified DD approach to finding a representative image region for each label, and then use the representative regions to annotate test image under Bayesian framework, at both image level and region level.

2.2.5 Discussion

Statistical models are popular in information retrieval, including automatic image annotation as described previously. Generally, they annotate images by estimating the joint probability of an image and a set of words, the probability of words given an image, or the probability of words given a specific image region. Documents or words are ranked according to their probabilities. From the viewpoint of statistical risk, this ranking principle is optimal if probabilities are calculated perfectly (Ripley, 1996). One of the drawbacks of using statistical models is probably the computational cost of parameter

estimation, i.e. the learning process. Most researchers chose the expectation maximization (EM) algorithm for finding the parameters that achieve the maximum likelihood on the training set (Duygulu et al., 2002; Fergus et al., 2003; Blei and Jordan, 2003; Li and Wang, 2003). For example, as reported by Fergus et al. (2003), it took about 24-36 hours to learn the parameters in their experiments on a training set of 400 images.

Vector space approaches for image auto-annotation consider each image as a vector of terms, which can be textual or visual. All the vectors are put together to build a term by document matrix. By applying sub-space techniques, it is hoped the latent semantic structure of the data-set will be revealed. Compared with statistical methods, vector space methods give a more clear vision of how images, image regions and words are related to each other, according to their positions in the feature space. However, the choice of the dimensionality of the sub-space is still an open question in many researches. In addition, when the data-set is very large, building a term by document matrix may cause some storage problems.

Classification approaches consider the process of annotation as finding the membership(s) of an image given a number of pre-defined categories of images. Empirically, SVM keeps good classification performance in many cases. For example, in chapter 4, our SVM based auto-annotation method achieves comparable results with the state of the art methods. A number of published methods which incorporate SVM also achieve very promising results on image classification or auto-annotation. Chen et al. (2006) used SVM in their MIL framework, and reported very high accuracy and fast computation in terms of image classification and object recognition. Classification methods, however, neither formally measure the probabilities of words related to images, nor organize the data in a way that is easier for users to understand.

2.3 Evaluation of Annotation Effectiveness

Once the test images are labeled by auto-annotation systems, annotation qualities need to be assessed for performance comparisons between different systems. A number of evaluation metrics have been used by researchers, some of which are introduced in the following.

2.3.1 Precision and Recall

Precision and recall, which are the most popular metrics for comparing different information retrieval systems, are also widely adopted for evaluating the effectiveness of auto-annotation approaches. In the information retrieval community, precision of a query is defined as the ratio of the number of relevant documents that are returned by the system to the total number of documents returned, and recall is defined as the ratio

of the number of relevant documents returned to the total number of relevant documents in the database. However, in the image auto-annotation community, precision and recall are defined slightly differently. There are two versions, per-image based and per-word based.

2.3.1.1 Per-image Precision and Recall

Per-image precision and recall are calculated on the basis of a single test image. For each test image, precision is defined as the ratio of the number of words that are correctly predicted to the total number of words predicted, and recall is the ratio of the number of words that are correctly predicted to the number of words in the ground-truth or manual annotations. Mathematically, they are calculated as follows

$$\begin{aligned} \text{Per Image Recall} &= \frac{r}{n} \\ \text{Per Image Precision} &= \frac{r}{(r + w)} \end{aligned} \tag{2.1}$$

where:

- r : the number of correctly predicted words;
- n : the number of manual labels in the test image;
- w : the number of wrongly predicted words.

Per-image precision and recall values are averaged over the whole set of test images to generate the *mean per-image precision and recall*.

2.3.1.2 Per-word Precision and Recall

Duygulu et al. (2002) used *mean per-word precision and recall* to evaluate their annotation effectiveness. It is also adopted by many other researchers (Jeon et al., 2003; Lavrenko et al., 2003; Feng et al., 2004; Carneiro and Vasconcelos, 2005; Yavlinsky et al., 2005) for comparison purposes.

Per-word precision and recall are calculated on the basis of each keyword in the vocabulary. Specifically, precision is defined as the number of images correctly predicted with a given word, divided by the total number of images predicted with this word. Recall is defined as the number of images correctly predicted with a given word, divided by the total number of images having this word in its ground-truth or manual annotations. The values are averaged over the words in the vocabulary to generate the *mean per-word precision and recall*. However, this metric is also not faultless, as will be described in Section 4.4.

2.3.2 Keyword Number with Recall>0

Duygulu et al. (2002) also used the *keyword number with recall>0* to show the diversity of correct words that can be predicted by the auto-annotation method. A keyword has recall>0 if it is predicted correctly once or more, otherwise not.

2.3.3 Normalized Score

Barnard et al. (2003) proposed to use *normalized score*, which is computed as the following equation, for auto-annotation evaluation:

$$E_{NS} = \frac{r}{n} - \frac{w}{N - n} \quad (2.2)$$

where:

r, n, w : the same definition as that of Equation 2.1;

N : the number of words in the vocabulary.

It equals 1 if the image is annotated exactly correctly, -1 if the exact complement of the actual word set is predicted, and 0 for predicting everything or nothing. However, this metric is not without problems. As argued by Hare and Lewis (2005c), according to the result reported by Monay and Gatica-Perez (2003), the normalized score is maximized when a very noisy result is generated, which is not desired.

2.4 Summary

In this chapter, we have mainly reviewed the background of this thesis. It began with a brief introduction to the research on CBIR. Three kinds of retrieval scenarios are introduced, namely category search, target search and association search. A few limitations of the traditional CBIR systems were addressed, the semantic gap problem in particular. Then, we went on to discuss automatic image annotation techniques, which are relatively new to the image community and seem to be a promising alternative to traditional CBIR. Three main categories of approaches have been reviewed, statistical approaches, vector space related approaches and classification approaches. In the end, we introduced several performance evaluation metrics, among which precision and recall are the most popular.

Chapter 3

Image Description

Digital images are generally considered as two dimensional matrices. Before being analysed by automatic annotation or object detection algorithms, they need to be condensed from pictorial information into feature values, so that the information which is important to the problem being solved can be retained or maximised while the redundancy can be removed. Image description is the process of generating descriptions that represent the visual content of images in a certain manner, normally in the form of one or more features.

We view the process of description as consisting of three steps, region choosing, feature extraction and feature quantisation, as shown in Figure 3.1. Region choosing selects one or more regions from the images in the spatial domain. Feature extraction condenses pixel values of each region into feature values. Feature quantisation maps feature values from continuous space into a discrete space. The first two steps exist in most cases, while the third step may be omitted depending on specific approaches.

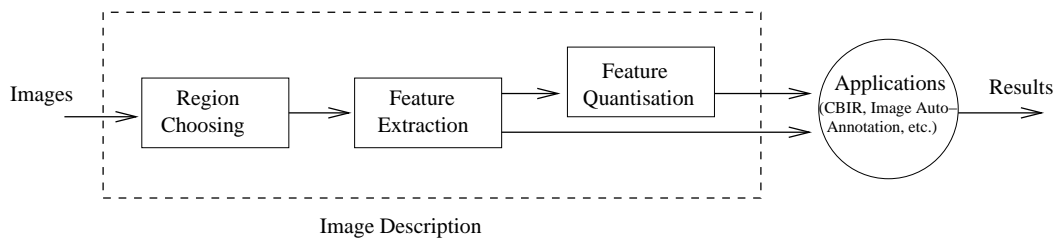


FIGURE 3.1: Three steps of the process of image description.

3.1 Region Choosing

In the early years of research on CBIR, global descriptors were the main choices for image description. However, different approaches (i.e. local descriptors) have been proposed later as researchers started to realise the limitations of global descriptors, especially

for applications where a particular object in the image is of interest. Local description approaches often choose parts from the images firstly, and then calculate descriptors for each individual part. Region choosing can be grouped into three categories, fixed partitioning, segmentation and saliency. Each category is described in the following. In fact, global description can be considered as a special case of region choosing, where the entire image is chosen as the region for feature extraction.

3.1.1 Fixed Partitioning

Fixed partitioning is the most convenient but naive form of region choosing. It applies the same division to each image, regardless the visual information that differs from one image to another. For example, Monay and Gatica-Perez (2003) divide images into three fixed regions, which are the image center, the upper part and the lower part, as shown in Figure 3.2(a). Qi and Han (2007) adopted a similar approach. The feasibility of this form of partition relies on the fact that many images depict the main objects in their center. However, this may not always hold. Some researchers (Lavrenko et al., 2003; Feng et al., 2004) simply divide images into tiles (rectangles) that have the same size, as shown in Figure 3.2(b).

3.1.2 Segmentation

Segmentation aims to partitioning images into object-shaped regions, each pixel of which belongs to one object in the real world and nothing else. Since manual segmentation by users costs a lot of effort and time, many automatic segmentation techniques have been developed over the years, such as the Normalised Cut (Shi and Malik, 2000) and JSeg (Deng et al., 1999). Figure 3.2(c) and 3.2(d) are two examples of a segmented image generated by automatic segmentation. Unfortunately, none really solves the problem of relating the segmented region to the actual object. Russell et al. (2006) use a multiple segmentations approach to find more accurate object extent based on the topics, or object classes, of the images found in the first place. This is consistent with the argument that the automatic image segmentation problem is “not just a bottom-up image processing problem, but also a top-down problem that requires knowledge of the true object” (Hare and Lewis, 2005c). Matthews et al. (2002) described both top-down and bottom-up approaches to visual feature extraction for lipreading. They implied that the combination of the two approaches are likely to improve on either of them. However, tailored segmentation algorithms may achieve fairly good results in selected narrow domains, in which the object is recorded against a clear background, such as trademarks.

3.1.3 Saliency

Salient points in an image are those that have special properties which make them stand out in comparison to neighbouring points. They are required to survive longest after different kinds of image transformations, such as scaling, rotation, blurring, addition of noise, viewpoint changes, illumination changes and so on. In other words, salient points detected in an image should also be detected after transformations. A compact image description can be derived based around the local attributes of the salient points. Much work has shown that saliency performs much better than some global image descriptors in terms of CBIR (Hare and Lewis, 2004; Sebe et al., 2002) and object recognition (Lowe, 1999). A number of different salient points detection methods have been suggested. In particular, using peaks in a multi-scale difference-of-Gaussian pyramid (Lowe, 1999, 2004) is a popular approach, as shown in Figure 3.2(e).

3.2 Feature Extraction

In the second step, features are extracted out of the image regions which are chosen from the first step. As mentioned previously, features can be global or local. When the region is chosen to be the whole image, features are global, describing the whole image. When the region is chosen to be a partition, segment or salient region, features are local, describing individual parts of the image. Features can also be categorised as being general or domain-specific. General features include commonly used features such as color, shape and texture. However, for special applications such as fingerprint recognition and lipreading (Matthews et al., 2002), general features are not applicable, so domain-specific features have to be developed. There are a great number of features used by researchers. MPEG-7 (Martinez, 2004) standardised a number of visual descriptors which are selected from many descriptors of a similar kind, through a strict evaluation procedure, and so are recommended as of high performance. Theoretically, the combination of different kinds of features will produce a more robust image description. In the following, some different image features used for CBIR and image auto-annotation are discussed, along with the details of those used in this thesis.

3.2.1 Colour

Colour is perhaps the most popular choice of visual features. It can be expressed in many different kinds of colour-spaces, such as the most widely used RGB space. RGB representation is in wide-spread use mostly because it describes an image in its literal colour properties.

The colour histogram, which can be calculated both globally and locally, is one of the most widely used colour descriptors. It is calculated by discretising the colour space

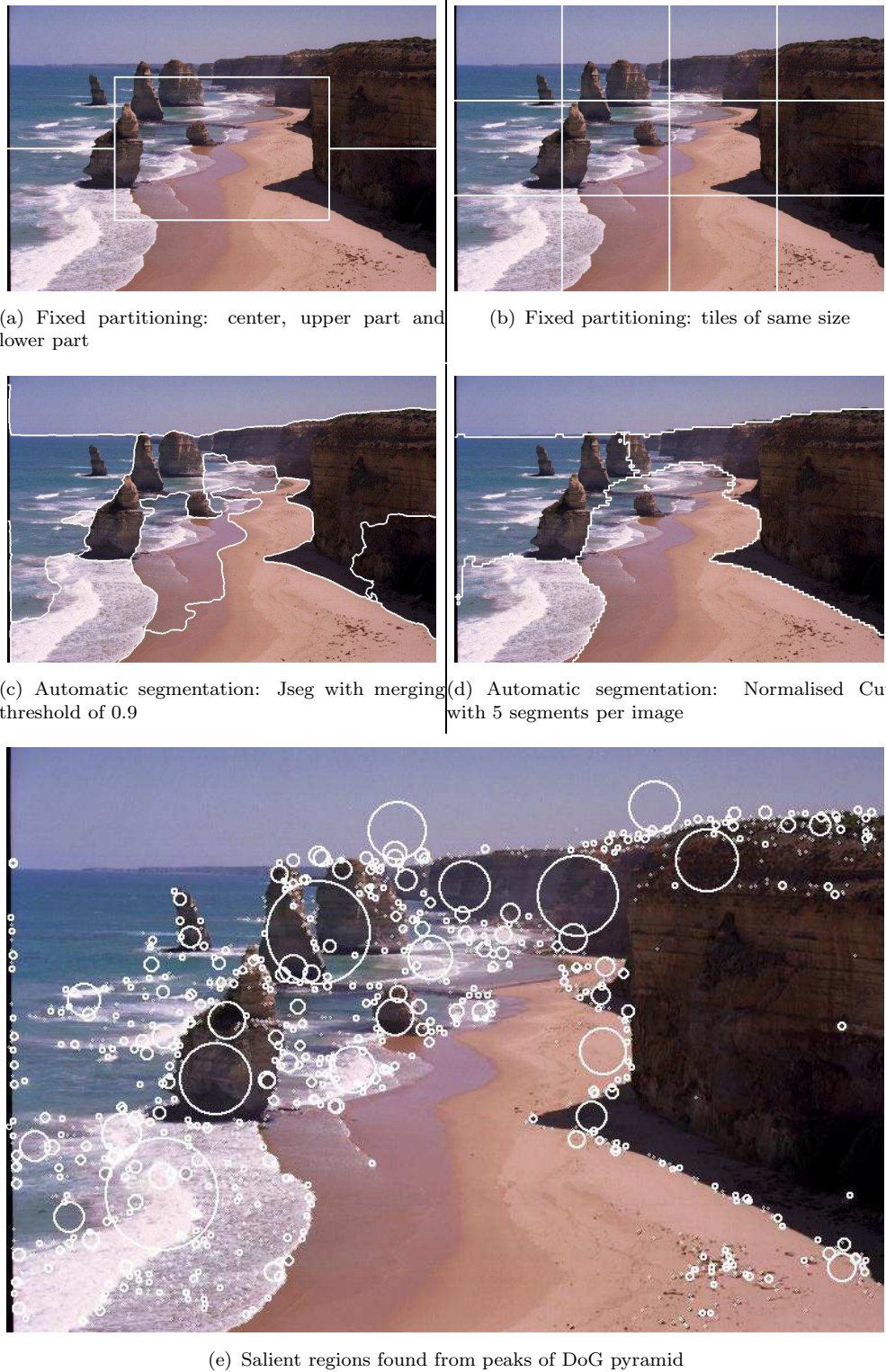


FIGURE 3.2: Different forms of region choosing for image description.

first and then counting the number of occurrence of each discretised colour in the image. Because histograms are accumulated over the whole image or region, with no information about locations, they are obviously invariant to translation and rotation of objects. The first use of colour histogram for image retrieval was proposed by Swain and Ballard (1991). They also argue that colour histograms are robust to change of viewpoint, scale and occlusion.

3.2.1.1 Colour Invariants

Although RGB colour has been found to be useful for a wide range of computer vision problems, cautions need to be taken when using it. In many applications, colour is assumed to be capable of capturing the intrinsic property of the imaged objects. However, this assumption is not totally valid, because the recorded colours of an image do not depend only on the objects but also on many other imaging conditions, such as illumination. RGBs are not stable across a change of illumination, and as a result applications based on raw RGBs could deliver poor performance. In order to solve this problem, a number of *colour invariant* approaches have been published in the literature. Colour invariants try to capture the features from colour that do not change with a change of illumination. Finlayson and Schaefer (2001) compared a number of colour invariant methods for image indexing. They showed that the colour invariant techniques were built for removing illumination dependency, but ignored the dependency on the capturing device, which is equally important. Different capturing devices have different sensors that may generate very different RGB responses. Finlayson et al. (2005) proposed a very simple but effective colour invariant image representation that is both illumination independent and also device independent. They applied histogram equalisation to each of the RGB channels of an image independently. This approach relies on the rank ordering of sensor responses that are preserved in practice for a wide range change of illumination and capturing devices.

3.2.1.2 The MPEG-7 Colour Structure Descriptor

The MPEG-7 [Martinez (2004)] Colour Structure Descriptor (CSD) is a colour descriptor that encodes information not only about the frequency of occurrence of colours, but also about their spatical layout in the image. It has been used in the experiments described in chapter 4.

An 8x8 structuring element is used to visit all locations in the image in order to compute the CSD. Suppose the colour space is quantized into M colours, a colour structure histogram is constructed such that the value of the i -th ($i = 0, 1, \dots, M-1$) bin represents the number of locations where the i -th colour exists in the structuring element. Although the number of samples is fixed to be 64, the spatial extent of the structuring element

changes with the the image size, and is determined by the following rule [Manjunath et al. (2001)]:

$$\begin{aligned} p &= \max\{0, \text{round}(0.5 * \log_2(\text{height} * \text{width}) - 8)\} \\ K &= 2^p, E = 8 * K \end{aligned} \quad (3.1)$$

where:

- $\text{height}, \text{width}$: image height and width;
- $E \times E$: spatial extent of the structuring element;
- K : sub-sampling factor.

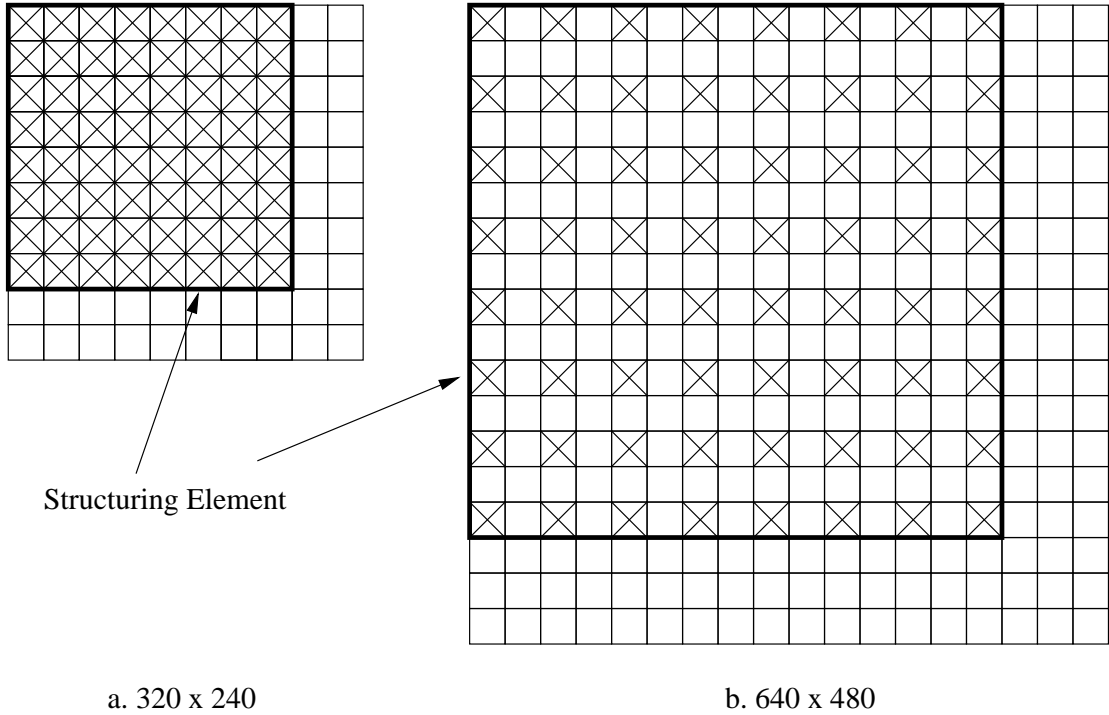


FIGURE 3.3: Structuring elements for images with different size (only a part of the image is shown)

As shown in Figure 3.3, for image size 320x240, no sub-sampling is used. However, for 640x480 ($p = 1, K = 2$ and $E = 16$). the extent of the structuring element is 16x16 and every alternate sample (denoted by ‘x’) along the rows and columns of the structuring element is used to calculate the histogram.

The CSD uses the HMMD colour space [Lee et al. (1998)], which contains 4 components - *Hue*, *Max*, *Min* and *Diff*. The transformation from RGB colour space to HMMD is conducted by Equation 3.2

Component	Subspace	Number of quantization levels for different numbers of histogram bins			
		256	128	64	32
Hue	0	1	1	1	1
	1	4	4	4	2
	2	16	8		
	3	16	8	8	4
	4				
Sum	0	32	16	8	8
	1	8	4	4	4
	2	4			
	3	4	4	2	1
	4			1	

TABLE 3.1: HMMD Colour Space Quantization for CSD

$$\begin{aligned}
Max &= \max(R, G, B) \\
Min &= \min(R, G, B) \\
Diff &= Max - Min \\
Hue &= \begin{cases} \frac{G-B}{Max-Min} * 60 & \text{if } (R = Max \wedge (G - B) > 0) \\ \frac{G-B}{Max-Min} * 60 + 360 & \text{if } (R = Max \wedge (G - B) < 0) \\ (2.0 + \frac{B-R}{Max-Min}) * 60 & \text{if } (G = Max) \\ (4.0 + \frac{R-G}{Max-Min} * 60) & \text{if } (B = Max) \end{cases} \quad (3.2)
\end{aligned}$$

Before calculating the CSD, the HMMD colour space is quantized into a number of bins. Four levels of quantization are defined, quantizing the colour space into 32, 64, 128 and 256 bins respectively. Taking the generation of 256 bins as an example, the quantization works as follows. First, the whole HMMD colour space is divided nonuniformly into 5 sub-spaces (0, 1, 2, 3 and 4). The division is performed on the value of *Diff* by the following intervals: [0, 5], [6, 19], [20, 59], [60, 109] and [110, 255]. Second, for each sub-space, uniform division is performed on the *Hue* and *Sum*, where $Sum = (Max + Min)/2$. The number of quantization levels for each sub-space is shown in Table 3.1. Finally, the bin values are divided by the total number of locations the structuring element reached and then normalized to 8 bits/bin, valued from 0 to 255.

3.2.2 Shape

Shape features describe the silhouettes of objects, so it requires the objects to be segmented out firstly. As we have mentioned, automatic segmentation techniques are still immature nowadays, the effectiveness of using shape descriptors in general CBIR or image auto-annotation applications is limited. However, shape plays an important role in some narrow domains such as trademark retrieval (Eakins et al., 1998) and recognition

of fish based on the shape (Mokhtarian et al., 1996), as a considerable amount of information was contained in the boundaries of objects. Two shape descriptors that have been used in the work of the thesis are described below.

3.2.2.1 Moment of Inertia

The Moment of Inertia has been used in the experiments in Section 7.2. It is defined as the rotational inertia with respect to the axis, which is perpendicular to the plane of the region and goes through the center of mass. Here is the equation (3.3) for each region:

$$Moment\ of\ Inertia = \frac{1}{N} \sum_{i=1}^N \left(\left(\frac{X_i - X_{center}}{l} \right)^2 + \left(\frac{Y_i - Y_{center}}{w} \right)^2 \right) \quad (3.3)$$

where N is the number of pixels on the boundary, (X_i, Y_i) are the coordinates of the i th pixel, (X_{center}, Y_{center}) are the coordinates of center of mass, (l, w) are the length and width of the region bounding box.

We normalized the distance from boundary pixels to the center of mass by dividing it by the size of bounding box, in the hope of decreasing the noise brought by the huge variance in region size. Moreover, the average moment is used for the same purpose.

3.2.2.2 The MPEG-7 Contour Shape Descriptor

The Contour Shape Descriptor is an MPEG-7 shape descriptor, based on the Curvature Scale-Space (CSS) representation of the contour. It is known as invariant to scaling, translation and rotation, which are very important characteristics of a good shape descriptor.

The CSS is a multi-resolution representation of the convex and concave sections of the contour at different scales. It captures inflection points (i.e. points at which curvature is zero) at each stage of the process of smoothing the contour progressively. The process of creating the CSS descriptor of a contour is as follows [Bober (2001)] :

1. Select N equi-distant points, starting from an arbitrary one, out of the contour so as to represent it.
2. Group the x, y coordinates of the N points into two series X and Y respectively.
3. Smooth the contour. In other words, apply a low-pass filter with kernel (0.25 0.5 0.25) to X and Y respectively.
4. Find inflection points, and then mark them on the so-called CSS image [Figure 3.4], where horizontal coordinates correspond to the indices of the contour points and vertical coordinates correspond to the counts of application of the filter.

Field	Number of bits	Meaning
Num of Peaks	6	Number of peaks in the CSS image
GlobalCurvature	2*6	Circularity and eccentricity of the contour
PrototypeCurvature	2*6	Circularity and eccentricity of the smoothed (convex) contour
HighestPeakY	7	Absolute height of the highest peak (quantized)
peakX[]	6	X-position on the contour of a peak (quantized, relative to the highest, clockwise)
peakY[]	3	Height of the peak (relative to the previous peak's height and quantized)

TABLE 3.2: Components of Curvature Scale-Space Descriptor [Manjunath et al. (2002)]

5. Go back to step 3 if inflection points still exist, otherwise stop.

Figure 3.4 depicts the CSS image (right column) of a contour, and its shape after 20 and 80 iterations (left column).

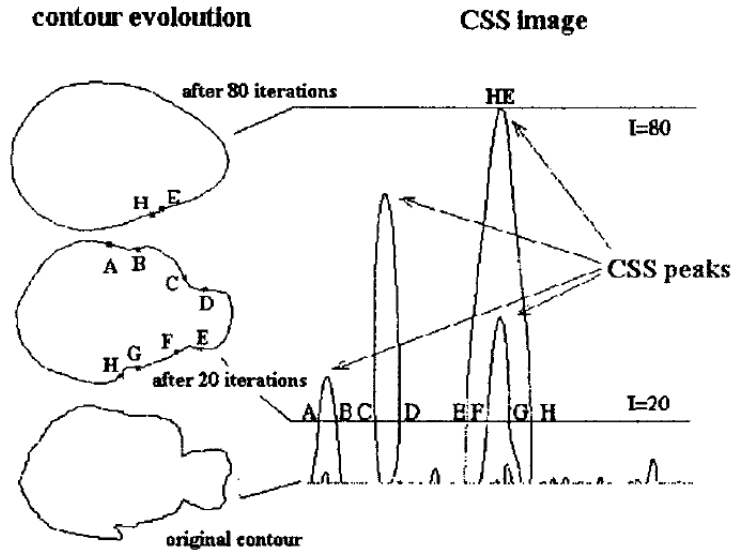


FIGURE 3.4: Example of contour evolution and corresponding CSS [Bober (2001)]

The Contour Shape Descriptor consists of several components [Mokhtarian and Bober (2003)].

$$D = \{E_c, C_c, E_p, C_p, yp_0, \{(xp_i, yp_i)\}; i = 1, \dots, k\} \quad (3.4)$$

where E_c, E_p and C_c, C_p are the eccentricities and circularities of the original and filtered (convex) contour respectively, yp_0 is the height of the highest peak (if contour is concave) and $\{(xp_i, yp_i); i = 1, \dots, k\}$ is a set of remaining peaks (possibly empty). See Table 3.2 for Details.

The definition of similarity measure of this descriptor is included in the MPEG-7 Standard, which is informative only. Users are free to choose different ways of measuring the distance between descriptors, in order to fit it to the characteristics of their class of shapes. Experimentation on the clustering of the Contour Shape Descriptor using Self-Organizing Map (SOM), which uses Euclidean distance as the similarity measurement, is presented in Section 3.3.1.2.

3.2.3 Texture

The definition of texture is vague, as Tuceryan and Jain (1993) said "we recognize texture when we see it but it is very difficult to define". They listed some example definitions attempted by different researchers. We understand texture as homogeneous visual patterns in images that manifest some kind of coherence or periodicity, such as wallpaper and bricks.

3.2.3.1 Mean Oriented Energy

The Mean Oriented Energy is a texture feature descriptor. It is computed by applying oriented filters to the image. Gabor filters are widely used oriented filters. "Properly tuned Gabor filters, can remove noise, preserve the true ridge and valley structures, and provide information contained in a particular orientation in the image." [Jain et al. (1999)]. In our experiment in Section 7.2, we chose the even symmetric Gabor filter, which is presented by Jain et al. (1999). 12 different oriented Gabor filters with 15 degree increments of θ , which is the orientation of the filter, are used. These filters are tested on the Camera-Man image. As shown in Figure 3.5, edges that share the same direction with the oriented filter are highlighted, otherwise removed.

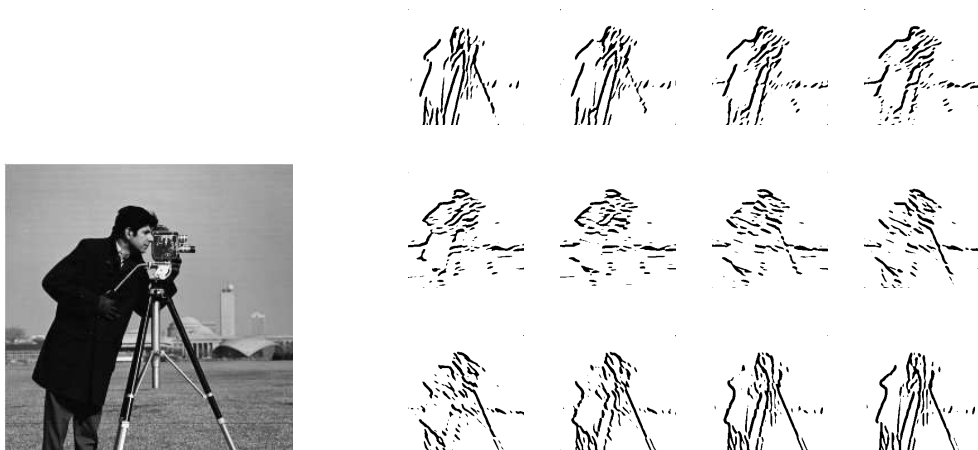


FIGURE 3.5: The original Camera-Man image and the result of being convoluted by 12 different oriented filter

Each image region is filtered by the oriented filters. The feature value of that filtered region is represented by the average absolute deviation from the mean, defined as

$$V_{\theta} = \frac{1}{N} \sum_{i=1}^N |F_{i\theta} - P_{\theta}| \quad (3.5)$$

where $\theta \in \{0^{\circ}, 15^{\circ}, \dots, 165^{\circ}\}$, N is the number of pixels in the region, P_{θ} is the mean of pixel values $F_{i\theta}$. Thus, each region gets a 12 dimensional feature vector that describes the texture.

3.2.4 SIFT - a local descriptor for saliency

For image description, approaches using salient points usually calculate a local descriptor for each salient point based on the pixel information of its neighbouring area, known as salient region. A great number of different local feature descriptors have been proposed for describing the content of a salient region. Mikolajczyk and Schmid (2005) compared several local descriptors and showed that SIFT (Scale Invariant Feature Transform) based descriptors (Lowe, 1999, 2004; Ke and Sukthankar, 2004) perform best.

The SIFT descriptor (Lowe, 2004) is designed to be invariant to image scaling, translation, and rotation, and partially invariant to change in illumination and 3D camera viewpoint. It encapsulates the information on gradient magnitude and orientation at each salient region. Lowe suggests that gradient location be quantised into a 4×4 location grid, and orientation be quantised into 8 orientation bins. This generates a 128 dimensional descriptor.

Ke and Sukthankar (2004) proposed to apply the Principal Component Analysis (PCA) technique to the descriptor representation and showed that their so-called PCA-SIFT descriptor is more distinctive and robust, and compact than the standard SIFT descriptor.

3.3 Feature Quantisation

Feature descriptors generated from the first two steps of image description, i.e. region choosing and feature extraction, can be processed directly by some applications for the problem to be solved. A very simple example is CBIR using global image features such as colour histograms which are represented as vectors. The similarity of two images is measured by the similarity of the corresponding vectors, which can be further calculated in a number of ways such as cosine distance and Euclidean distance. Given a query image, all the images in the database are ranked according to their distances to the query.

However, for some other applications, the third step - feature quantisation, needs to be applied. One example is applications where saliency is used for image description. The number of salient points found in images can be very large. For example, the number of salient points found from the Washington set images (University of Washington, 2004), which have an average resolution of 640×480 , is in general several thousand per image, using the “difference-of-Gaussian pyramid” approach. It is not convenient for image retrieval or auto-annotation algorithms to process so many salient points per image directly, especially when each point is represented by a high dimensional feature descriptor. Feature quantisation is a process of grouping similar image feature descriptors into the same class and different ones into different classes. As a result, images can be described by the membership, a single number, of descriptors instead of the actual high dimensional values. Feature quantisation can also be regarded as a classification problem in which the membership of each feature is to be determined. In the following, two clustering techniques are discussed, namely k -Means Clustering and the Self-Organizing Map (SOM).

3.3.1 The Self-Organizing Map (SOM)

The Self-Organizing Map (SOM) is a neural network-based data visualization tool invented by Professor Teuvo Kohonen. “It converts complex, nonlinear statistical relationships between high-dimensional data items into simple geometric relationships on a low-dimensional display. The SOM usually consists of a two-dimensional regular grid of nodes.” [Kohonen]. Similar data items should be organized closer than more dissimilar ones. Reducing dimensions and displaying similarities are the two valuable characteristics of this technique.

3.3.1.1 The SOM Toolbox

The SOM toolbox¹ is a function package developed for Matlab to implement the Self-Organizing Map algorithm. A SOM consists of neurons organized on a regular low-dimensional grid [Vesanto et al. (2000)]. Each neuron represents a weight vector which has the same dimensions as the data set to be visualized (i.e. the input data set). The final SOM for visualizing the high dimensional data set is obtained by training iteratively (maybe several hundred times). The idea is that in each training step, not only the Best-Matching Unit (BMU, the neuron whose weight vector is closest to the input sample, which is picked from the input data set) but also its neighbors are updated: the region around the BMU is stretched towards the training sample, Figure 3.6 [Vesanto et al. (2000)]. In the end, neurons on the map become ordered: neighboring ones have similar weight vector.

¹Available at: <http://www.cis.hut.fi/projects/somtoolbox/>

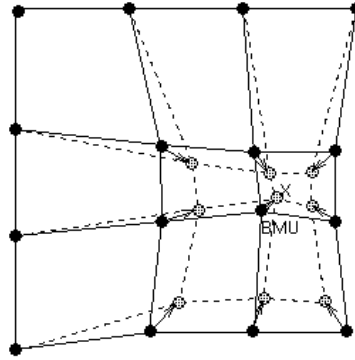


FIGURE 3.6: Updating the best matching unit (BMU) and its neighbors towards the input sample marked with x. The solid and dashed lines correspond to situation before and after updating, respectively [Vesanto et al. (2000)]

3.3.1.2 Shape Clustering Using CSS and SOM

The Curvature Scale-Space (CSS) (3.2.2.2) descriptors of 1100 marine creature shape images², which are used in the SQUID system³, are extracted. These 1100 descriptors are then clustered by SOM into 155 (11x15) clusters. Figure 3.7 are three random cells (clusters) from the SOM, each of which is represented by 6 sample shapes from it. It shows that the shapes are well clustered. Besides, one shape from each cell of the SOM is taken to construct the SOM, in order to give a visual overview of the whole SOM, as shown in Figure 3.8. Thanks to the clustering characteristics of SOM, as described in section 3.3.1, shapes which are similar but not similar enough to be clustered within the same cell are placed as neighbours.

Cell A	Cell B	Cell C

FIGURE 3.7: Three random cells from the SOM of 1100 marine creature shapes

²Available at ftp://ftp.ee.surrey.ac.uk/pub/vision/misc/fish_contours.tar.Z

³<http://www.ee.surrey.ac.uk/Research/VSSP/imagedb/demo.html>

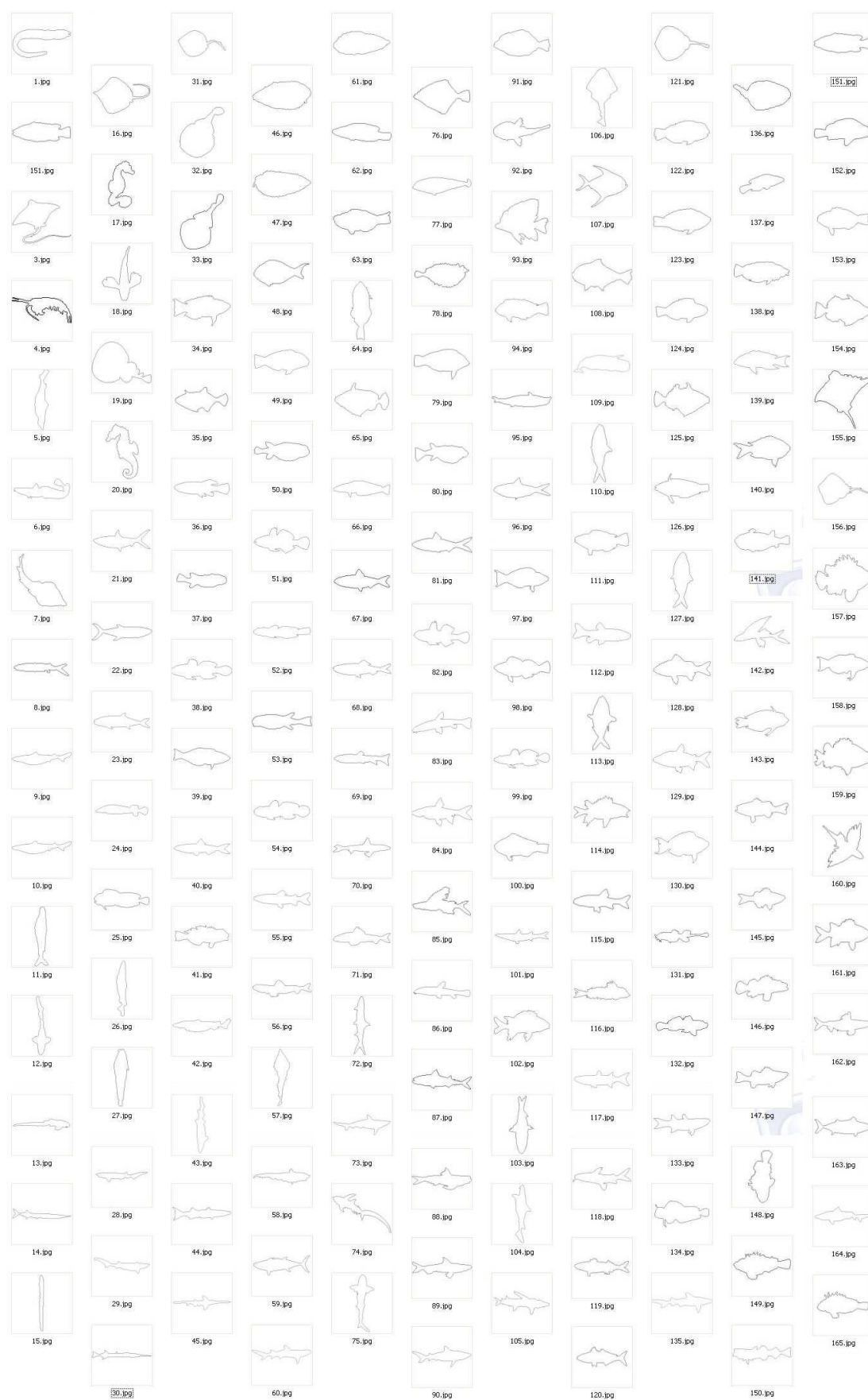


FIGURE 3.8: The SOM of 1100 marine creature shapes

3.3.2 k -Means Clustering

The k -Means is an algorithm to cluster objects, or data points, into k partitions, or clusters. The clusters are discovered through an refinement process that updates the position of clusters iteratively. During each iteration, all the training points are assigned to the closest cluster based on the distance to the cluster centroid. Then, the centroid of each cluster is updated by the new cluster centroid which is calculated as the centroid of all the points that belong to it. The process is repeated until the points no longer switch clusters, or after a pre-defined number of iterations. Decisions on the value of k and the starting cluster centroids are essential to the performance of k -Means. A common choice of the initial centroids is to choose k sample points at random and use them as the centroids.

The k -Means is one of the most popular techniques for multi-dimensional vector quantisation in image description. For example, Duygulu et al. (2002); Jeon et al. (2003) use it to quantise the global feature descriptors of image segments and then represent each segment with the membership of the descriptor. Hare and Lewis (2004) use it to quantise SIFT descriptors of salient regions found in images and then represent each image as a histogram of the membership of descriptors.

3.4 Image Description Examples

In this section, we describe two concrete examples of image description which have been adopted in the experiments of the thesis. The first form is the “blob” representation used in the work of Duygulu et al. (2002), and the second form is visual term representation used in the work of Hare and Lewis (2004).

3.4.1 The “blob” representation

The first form of description uses segmentation for region choosing. Images are described as consisting of several “blobs”. The process is conducted as follows.

1. Images are segmented into object-shaped regions using automatic image segmentation algorithms. Ideally, each image segment or region contains an object or object part. Very small regions with an area that is below a certain pre-defined threshold are discarded, because such regions are considered as too small to contain the objects we are interested in and may bring noise to the system.
2. A descriptor is calculated for each image segment, typically using colour, shape and texture features.

3. the whole set of descriptors are quantised into classes in order to group segments of the same object into the same class. Each class is called a “blob” by Duygulu et al. (2002).
4. Finally, each image can be represented as a collection of blobs.

Each step has a major influence on the final description effectiveness. Intuitively, faulty segmentation makes it unlikely for auto-annotation algorithms to learn the true characteristics of objects. An advanced segmentation method that is able to find the accurate object boundaries is important to the whole annotation process. A suitable image description makes segments with different object more distinguishable from each other. A robust clustering method is more likely to discover the true membership of segments. Figure 3.9 depicts the process of generating “blobs”. Ideally, segments with the same object are assigned to the same “blob”. As can be seen, “bear” belongs to “blob” 1 and “pyramid” belongs to “blob” 2, and so on.

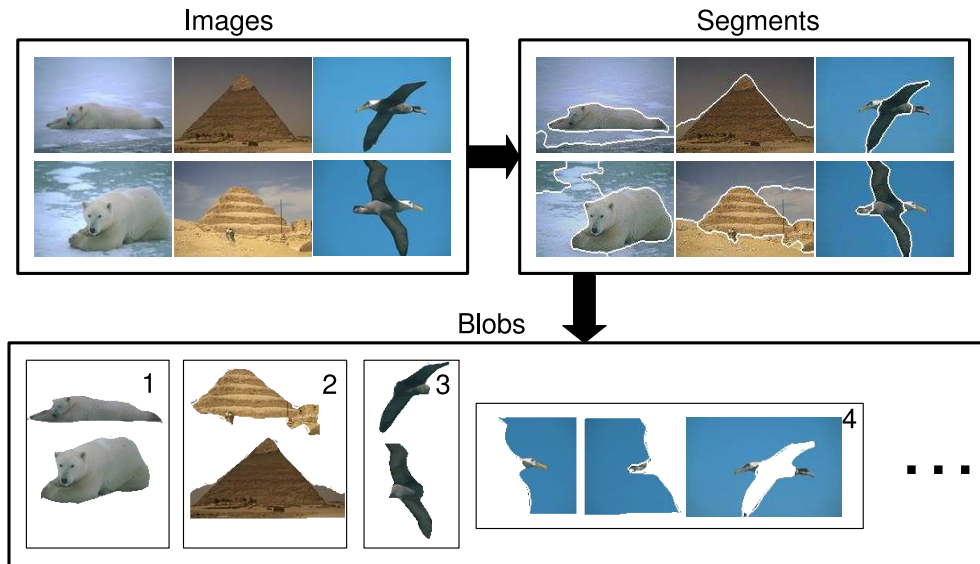


FIGURE 3.9: The process of generating “blobs” for image description.

3.4.2 Saliency based visual term representation

The visual term representation has been adopted widely for image description (Hare and Lewis, 2004; Sivic et al., 2005). The process is generally conducted as follows. Figure 3.10 gives a diagram of the approach.

1. Salient regions are discovered by saliency detection techniques on each of the images from a data-set.
2. For each salient region, a local feature descriptor is calculated, for example the SIFT descriptor in the example shown in Figure 3.10.

3. Salient descriptors of all the images are quantised into clusters, each of which is regarded as corresponding to a visual term or visual word. Generally, the centroid of each cluster is chosen as the visual term. The whole set of visual terms constitute the so-called visual vocabulary. A salient region or salient point is then represented by a visual term indicating its membership of a cluster.
4. Finally, each image can be described as a histogram of visual terms, indicating the number of occurrence of each term in the image.

This form of description is analogous to the way in which a set of text words constitute a text document. Here, each quantised salient descriptor is considered as a word, and each image is a document.

Another slightly different form is to treat an image segment as a document (Russell et al., 2006), as opposed to the entire image. An automatic image segmentation algorithm is used to divide images into object-shaped segments. As a result, salient points are divided into groups according to their locations in the image. Each image segment is then represented by a histogram of visual terms, in the same manner as that for the entire images described above.

The performance of saliency based visual term description of images is also influenced by several factors, including the performance of the saliency detection algorithms, the effectiveness of the local descriptor for the salient regions, the quality of quantisation of descriptors and so on.

3.5 Summary

We have described three main aspects of image description, namely region choosing, feature extraction and feature quantisation. Since global image descriptions lack local information about images, region choosing is conducted to divide images into parts so that features can be extracted locally. Three forms of region choosing are fixed partitioning, segmentation and salient region. Once the regions are chosen, feature extraction is applied. There are a great number of different feature descriptors, from basic colour, shape, texture descriptors to more advanced local descriptors such as SIFT. In some cases, the third step is adopted to group feature descriptors into classes, so that CBIR or image auto-annotation systems can avoid dealing with a massive number of feature descriptors which are possibly of high dimension.

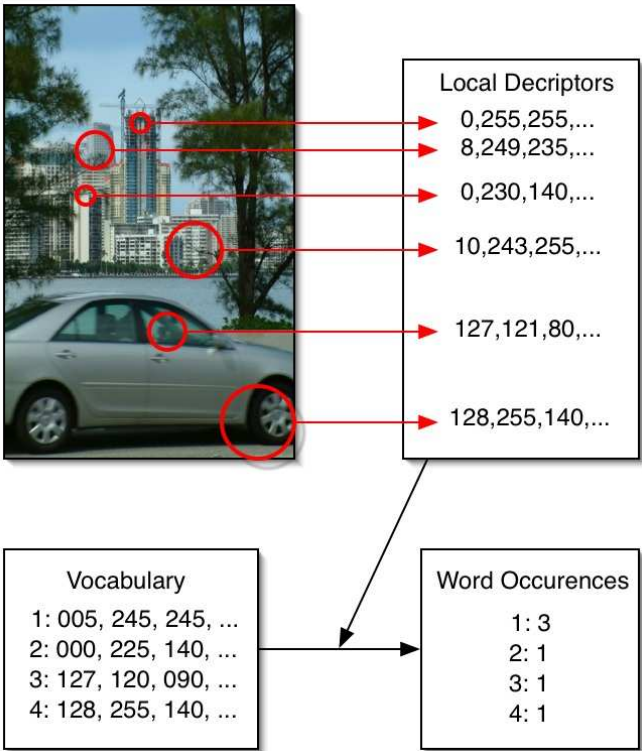


FIGURE 3.10: Image representation using quantised salient region descriptors (Hare, 2006).

Chapter 4

Quality Issues with Data-Sets

So far, researchers have been focusing primarily on developing various auto-annotation algorithms (Duygulu et al., 2002; Barnard et al., 2003; Jeon et al., 2003; Feng et al., 2004; Pan et al., 2004), but very few have examined the effect of the data-set itself on the annotation result. Although good annotation algorithms certainly really need to be advanced, the choice of appropriate data-sets for experiments is also important. An inappropriately designed data-set could give a biased measure of how well certain methods work. Westerveld and de Vries (2003) used the Corel images for evaluating their image retrieval technique and claimed that the Corel data-set is relatively easy. This issue was also confirmed by Viitaniemi and Laaksonen (2007), who investigated three benchmark image data-sets.

This chapter gives extensive and detailed experiments and discussions on quality issues of image data-sets in the context of automatic image annotation. We have developed and applied three auto-annotation methods to two image collections, one of which is built by capturing images from the web. Through the experiments and by comparisons of the results, we have examined several issues about image data-sets, including problems when training sets and test sets contain many very similar images and data-sets with redundant information.

This chapter is based the publications by the author in (Tang and Lewis, 2006) and (Tang and Lewis, 2007a). It begins with a description of two experimental image data-sets, followed by the details of the three annotation approaches used for comparison. Afterwards, it shows the results and discusses the quality issues related to data-sets for image auto-annotation.

4.1 Two Image Collections

4.1.1 The Corel Set

The first image set we consider is the widely used Corel Image set provided by (Duygulu et al., 2002) which is already separated into a training set with 4500 images and a test set with 500 images. Most of the images have 4 word annotations, while a few have 1, 2, 3 or 5. The vocabulary size of the whole set is 374 and that of the test set is 263. We note that in fact, the crucial vocabulary size is that of the training set since no other words are accessible for the auto annotation process. The vocabulary size of the Corel training set is 371. It will be shown in Section 4.4 that the simple colour structure descriptor (CSD)-based propagation method (Tang and Lewis, 2006) achieves good results, compared with some state-of-the-art methods, for this image set. We argue that this is indeed because the Corel images are relatively easy to annotate and that this in turn is a result of the presence of groups containing many closely related images in the collection. This can actually be seen clearly from the following analysis. In the Corel training set there are 2705 images with 4 word annotations and these comprise a vocabulary of 342 different words. Among the 2705 images with 4 word annotations there are only 1833 different combinations of words. Assuming only random selections, the probability of getting such a low number of different combinations in a sample of 2705 is almost 0 ($\sim 10^{-4796}$). Of course, some of the departure from randomness is due to the way in which some objects or image features appear frequently together in nature, for example trees and grass.

The fact that some of the images are close to each other in terms of both low-level features (such as color) and also the semantics and thus have the same combination of keywords for their annotations, is shown in Figure 4.1. A query image can be annotated correctly by a simple propagation method if there exists a training image that is very similar (both at the low-level and semantically) and if this is the one chosen for propagation.

4.1.2 The Yahoo Set

In order to create a new image set avoiding, if possible, groups of similar images, a new image collection was created by querying the Yahoo Image Search engine¹ using each of the 263 keywords from the Corel test set (Duygulu et al., 2002), such as ‘water’, ‘sky’ and ‘people’. For each keyword, the first 20 images returned by Yahoo are selected and annotated with the single query keyword used to retrieve it, resulting in a collection of 5260 images. All images are JPG color images, with a resolution of 120x80 on average. In some cases these annotations were not particularly appropriate because of the text based nature of the Yahoo image search. For example, image 4.2(a) is retrieved by using

¹<http://images.yahoo.com>



FIGURE 4.1: Examples of similar Corel images, the number in the parenthesis being the file name of the image.

the word “water”, probably because there was an article about drinking water around the image and the word “water” appeared so many times that the Yahoo image search assumed it was a “water” image. In addition, the images are sparsely annotated because most images have more than one object. For example, images 4.2(c) and 4.2(d) contain multiple objects, but are only annotated with a single word. It is a more challenging set because, unlike the Corel set, the collection is less likely to contain groups of images with very similar content. The implication is that effective training with the Yahoo set² will be more difficult than with Corel.

In order to illustrate the self-similarity problem of the Corel set, we computed on each data set (Corel and Yahoo) the Euclidean distance between each image and its nearest neighbour (NN) in the CSD feature space. As shown in Fig. 4.3, the X axis represents the value of distance, while the Y axis represents the number of images that have a NN at this distance. The average distance for the Corel set and Yahoo set are 210 and 241 respectively. Statistically, 23.5% of the Yahoo images have a NN with a distance less than the average value of the Corel images. In contrast, up to 73.4% of the Corel images have a NN with a distance less than the average value of the Yahoo image.

²Available at: <http://www.ecs.soton.ac.uk/~phl/YahooSet.tar.gz>

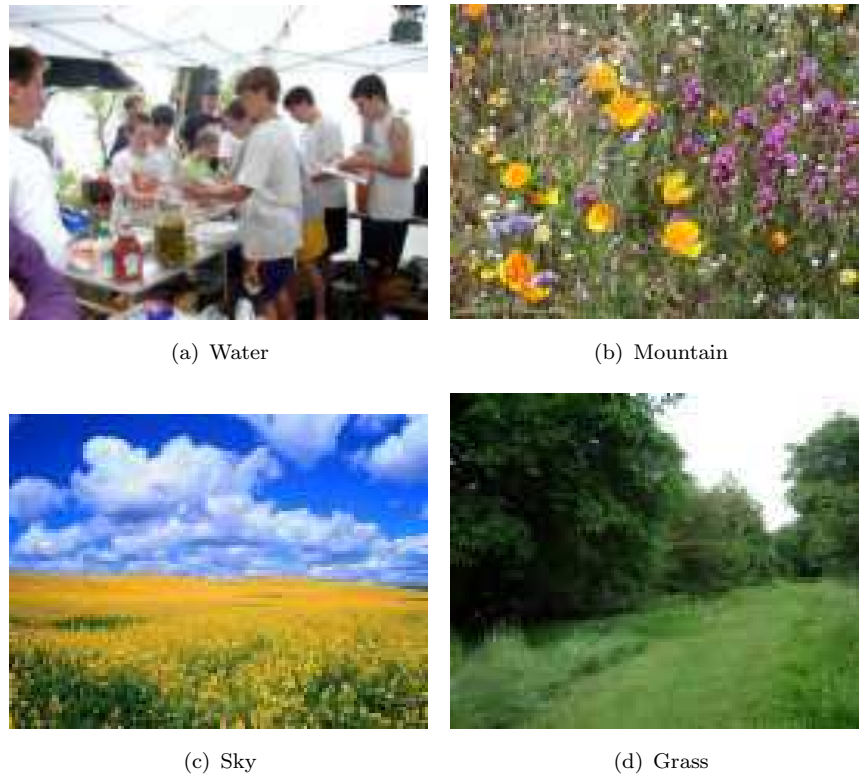


FIGURE 4.2: Examples of Yahoo images. The top images are inappropriately annotated, and for the bottom images only one object is annotated.

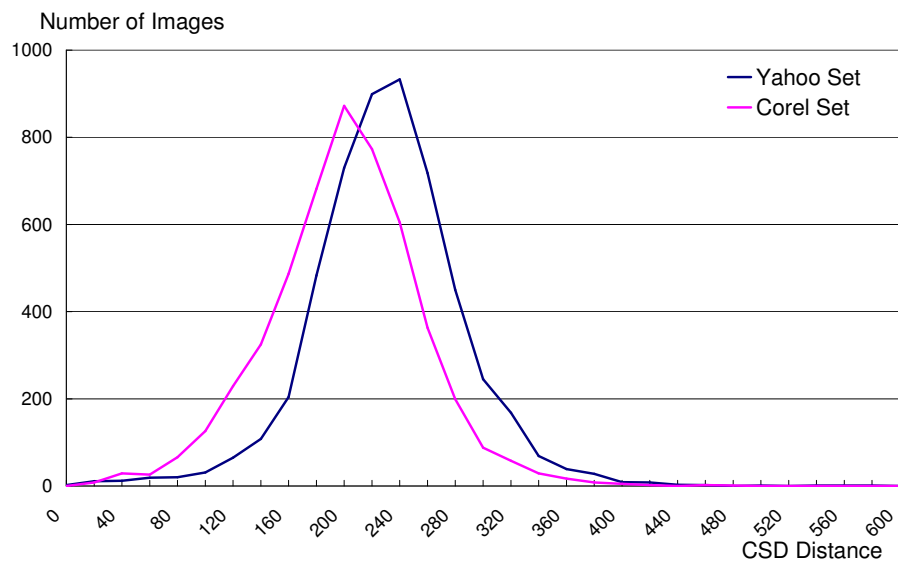


FIGURE 4.3: The curves show, on the Corel and Yahoo set respectively, the CSD Euclidean distance between each image and its nearest neighbour (NN).

Image Index	Captions		
1	a	b	c
2	b	d	e
3	a	b	d

TABLE 4.1: Illustration of propagation for the CSD-Prop method

4.2 Three Auto-annotation Methods

We have implemented and used three very different approaches to image auto-annotation for the main comparison of methods. These methods are designated as CSD-Prop, SvdCos and CSD-SVM. CSD-Prop (Tang and Lewis, 2006) is a propagation method based on a global feature vector, the MPEG-7 Colour Structure Descriptor (CSD) (Martinez, 2004). SvdCos is a more complex region based method using correlation statistics, based on the work of Pan et al. (2004). Finally CSD-SVM is a multi-class and multi-label image classification method.

4.2.1 The CSD-Prop Method

Propagation methods (Monay and Gatica-Perez, 2003; Hare and Lewis, 2005c) work by propagating annotations from the most similar images in the training set. In this work, the MPEG-7 Colour Structure Descriptor (CSD) (Martinez, 2004) is used as the feature descriptor to rank the training images. The similarity between images are measured by the CSD distance (squared euclidean). For each test image, propagation starts from the top training image and goes on until a desired number of different annotations are found. Because the number of predicted words for a test image is fixed, sometimes only a portion of the annotations of a training image can be used. When it is the case, the choice is made randomly. For example, if the top 3 training images in the ranked list for a test image have the captions as showed in Table 4.1 and 4 words need to be predicted, they are either ‘a’, ‘b’, ‘c’, ‘d’, or ‘a’, ‘b’, ‘c’, ‘e’.

4.2.2 The SvdCos Method

The region based SvdCos Method is proposed by Pan *et al.* (Pan et al., 2004) and uses the blob representation proposed by (Duygulu et al., 2002). Following (Pan et al., 2004)’s derivation, the SvdCos method works as follows. Suppose there are N_W words in the vocabulary and N_B blobs in the visual vocabulary, the whole training set $I = \{I_1, \dots, I_{N_I}\}$ can be represented by a matrix $D_{[N_I \times (N_W + N_B)]}$, where $D = [D_W | D_B]$. The (i, j) -element of D_W is the count of word w_j in image I_i , and the (i, j) -element of D_B is the count of blob b_j in image I_i . This method captures the association between words and blobs through their pattern of occurrence over the whole image set, which is represented

by each column of D_W and D_B . A translation table $T_{[N_W-by-N_B]}$ is created. T_{ij} is the cosine value of the i th column vector of D_W and j th column vector of D_B . Each column of T is normalized to add up to 1. Thus, T_{ij} can be treated as the probability of translation between word w_i and blob b_j .

Singular Value Decomposition (SVD) decomposes a matrix $X_{[n-by-m]}$ into a product of three matrices U , Λ and V^T , where U and V are orthonormal, and Λ is diagonal. Previous works (Furnas et al., 1988) show that by eliminating small diagonal values of Λ , “SVD could be used to clean up noise and reveal informative structure” ((Pan et al., 2004)) in X . Therefore, SVD is applied to D before constructing the translation table. Given a test image, which is represented by $q = \{q_1, \dots, q_{N_B}\}$ (where q_i is the count of blob b_i), it can be annotated by choosing the words that have the highest values in p , where $p = Tq$.

Details of this method can be found in (Pan et al., 2004).

4.2.3 The CSD-SVM Method

Image auto-annotation can be handled as a multi-class and multi-label classification problem. Multi-class means there are more than two classes, each of which is represented by a keyword, while multi-label means each image belongs to multiple classes. For example, images of 4.1(a) belong to the classes “Birds”, “Sea”, “Sun” and “Waves”. Multi-label classification is more difficult than single-label classification. In the following, we propose to turn the multi-label problem into a single-label problem, using the CSD-SVM method, and describe how optimal parameters can be found.

CSD-SVM is a method based on Support Vector Machines (SVM), which is a very popular technique for classification. Two common ways of multi-class classification by SVM are “one-vs-all” and “one-vs-one”. As for multi-class classification, we choose the “one-vs-one” method, which uses a voting scheme. A binary classifier is trained for each possible combination of two classes. The class with the highest votes for the test document wins. In order to predict multiple annotations for each test image, rather than using only the class with the highest vote, we use the top n classes that have the highest votes, n being the number of annotations to be predicted. Another issue is that each training image has multiple labels, which makes it a multi-label training problem. We turn it into a general single-label training problem by duplicating each training image based on the number of words it has, and assign one and only one of the words to each copy of the image. Therefore, when applying the “one-vs-one” method, there are cases in which some training documents belong to both classes.

We used LIBSVM (Chang and Lin, 2001) to classify images that are represented by the Colour Structure Descriptor (CSD) as used for the CSD-Prop method. The radial basis function (RBF) (Hsu et al.), is used as the kernel function. Two parameters need to

be optimized in LIBSVM, namely the penalty parameter C and the kernel parameter γ (Chang and Lin, 2001). As recommended in (Hsu et al.), a grid-search method is applied on C and γ to find the optimal values, using v -fold cross-validation. The pair (C, γ) achieving the highest cross-validation accuracy is finally used to classify the test documents. However, calculating the cross-validation accuracy is not as straightforward as that on general single-label training problems. Instead, for each training image being used for testing, we have to compare the predicted label with all the words attached to this image before the image is duplicated. If the predicted label is one of them, the image is regarded as being correctly classified.

4.3 Evaluation Metrics

The *Mean Per-word Precision and Recall* and *Keyword Number with Recall* > 0 , as used by previous researchers Duygulu et al. (2002); Jeon et al. (2003); Feng et al. (2004); Carneiro and Vasconcelos (2005), are adopted for evaluating annotation effectiveness. Per-word precision is defined as the number of images correctly annotated with a given word, divided by the total number of images annotated with this word. Per-word recall is defined as the number of images correctly annotated with a given word, divided by the total number of images having this word in its ground-truth or manual annotations. Per-word precision and recall values are averaged over the set of test words to generate the mean per-word precision and recall. A keyword has recall > 0 if it is predicted correctly once or more, otherwise not.

We also introduce *Mean Per-image Precision and Recall* and *Cumulative Correct Annotations* for evaluation. Per-image precision is the number of correctly predicted words for a given image divided by the total number of words predicted for that image, and per-image recall is the number of correctly predicted words divided by the number of manual annotations for that image. Per-image precision and recall are then averaged over all the test images to get the mean per-image precision and recall. Cumulative Correct Annotations is the total number of correct annotations.

4.4 Results and Discussion

4.4.1 Comparison with state-of-the-art methods

We applied the previously described annotation algorithms to the Corel set and predict 5 words for each test image. Table 4.2 compares the CSD-Prop, SvdCos and CSD-SVM methods with the results of some state-of-the-art methods taken from the literature when the Corel training set is trained to annotate the Corel test set. These methods are the Translation model (Duygulu et al., 2002), the CRM model (Jeon et al., 2003), the

MBRM model (Feng et al., 2004), and the Mix-Hier model (Carneiro and Vasconcelos, 2005).

It is interesting to note in Table 4.2 that our simple CSD-Prop method achieves results almost as good as the best results from the more advanced methods. We argue that this is due to the training set and test set containing very similar images as illustrated in Figure 4.1. For example, in our experiment, the CSD-Prop method successfully predicts the word “Kauai”, which is even unlikely for a human being to learn, for the test image 4.1(c). It results from the training set containing the image 4.1(d) which is the closest in terms of Euclidean distance in the CSD feature space.

The best performance for the CSD-SVM algorithm is found to be at ($C = 2^5, \gamma = 2^{-1}$) using grid-search and cross-validation. From the results, our CSD-SVM method performs reasonably well in the experiments when compared with the other methods considered. Although it gets a slightly lower number of words with recall>0 than the CSD-Prop method, overall the CSD-SVM achieves better results than CSD-Prop in view of the higher precision and recall measures. Again we argue that the reason why CSD-Prop method achieves a higher number of words with recall>0 is because it benefits from the fact that the Corel training set and test set have many globally very similar images in common. Difficult words, such as the place name “Kauai”, are easily learned by just comparing image similarity. The CSD-SVM method is also comparable in performance with the Mix-Hier method, which achieves the best results from the state of the art literature based methods in terms of mean per-word precision and recall.

4.4.2 An Examination of word combinations

In this section, annotation effectiveness is analysed further in terms of word combinations. We consider the combination of four words that are correctly predicted, since most of the Corel images have 4 ground-truth labels, and for a single test image the auto-annotation methods predicted a maximum of four correct words. The SvdCos method is excluded from the analysis as it only managed to get 4 words correct 5 times, which is much less than that of CSD-Prop (45 times) and CSD-SVM (53 times).

The analysis was conducted as follows on the predicted annotations both for CSD-Prop and CSD-SVM. Firstly, we find all different kinds of 4 word combinations from the predicted annotations on the test set, under the condition that each word is correct. Then, for each combination found, we search the training set to see if such a combination of annotations exists and if it does, how many times it occurs. Considering the propagation nature of CSD-Prop, it is not surprising that for CSD-Prop, all predicted 4 word combinations are found in the training set. For CSD-SVM, 51 out of 53 are found. The fact that almost all of the correct 4 word predictions exist in the training set, implies that this CSD based method may only be learning the relations between

the whole image and the corresponding word sets or object sets from the training set which is certainly the situation for the CSD-Prop method. For correctly predicted four word combinations, Figure 4.4(a) shows their number of occurrences in the training set and in the predicted annotations by CSD-Prop and CSD-SVM. Those with number of occurrences greater than 5 in the training set are shown in Figure 4.4(b). Note that for the last two combinations (Figure 4.4(a)), the number of occurrence in the training set equals zero. This means that CSD-SVM managed to predict two combinations that do not exist in the training set, as shown in Figure 4.5. In the training set, the word combinations “clouds, sun, water, tree” and “buildings, clothes, shops, street” do not exist, but CSD-SVM managed to predict them correctly. Moreover, it can be seen that the other words predicted by CSD-SVM predict, “palm” and “people” for images 1061 and 119088 respectively, are actually reasonable annotations though not in the ground truth. The words predicted by CSD-Prop are included for comparison. All in all, if image auto-annotation is recognised as a problem of object recognition, the relations between objects and words, rather than the whole image and words, are really what need to be discovered. A good annotation method should be able to predict object combinations in the test images, no matter how these objects occur in the training set, either together in single images or separately in different ones. The use of global descriptors in annotation algorithms severely limits the possibility of achieving this.

4.4.3 Comparison between the three methods when different training sets are used.

For each of the three methods, we used the Corel training set and the Yahoo set for training respectively, to annotate the Corel test set, each image being predicted by 5 words. However, for fair comparison, only one random word out of the complete set of captions (normally 4) is used for each Corel training image, since each Yahoo image has only one caption. For example, for Fig. 4.1(a), we randomly choose one of the words “Birds”, “Sea”, “Sun” and “Waves” as the only annotation for this image and discard the others. Note that the whole set of labels of the test images are kept for evaluation. Table 4.3 compares the three methods using the two different image sets for training.

It can be seen that the CSD-Prop method performs better than the SvdCos method when it is trained on the Corel training set, but worse than the SvdCos method when trained on the Yahoo set. In other words, the CSD-Prop method degrades more rapidly when it moves from an easy training set (Corel) to a more difficult set (Yahoo). The CSD-SVM method maintains relatively good performance in both cases.

It can be seen that, even though only about 25% of the annotations of the Corel training set are used, the results of these methods did not decrease as much when compared with those referred to in Table 4.2, where 100% of the annotations are used. The results of the CSD-SVM method are even comparable with that of the CRM (Jeon et al., 2003)

method, which uses 3 times more annotations. This implies redundant information exists in the Corel training set. For example, both image 4.1(a) and 4.1(b) belong to the training set. Since they are so similar to each other in both low-level features and semantics, there is little need for an annotation method to learn on both, especially for computationally expensive methods or when the training set is extremely large. Potentially, reduction techniques (Wilson and Martinez, 2000) can be used to condense the training set, reducing the size while retaining important training information.

We conclude that it is relatively easy to annotate the Corel test set using the Corel training set, and that the CSD-Prop method does not transfer as well as the SvdCos and CSD-SVM methods to the more challenging Yahoo dataset. In addition, it can be argued that using a challenging data set, good auto-annotation approaches should perform substantially better than, propagation-based approaches. Finally we conclude that simple sets like the Corel set should be used with caution for effective evaluation of annotation methods.

4.5 Summary

This chapter has demonstrated some of the disadvantages of data-sets like the Corel set for effective auto-annotation evaluation. The three image auto-annotation methods, CSD-Prop, SvdCos and CSD-SVM, have been used to annotate the Corel test set, by training on two different training sets, the Corel training set and the Yahoo training set. The Yahoo training set was constructed by obtaining images from the Yahoo Image Search Engine for 263 query words.

Through the experiments described above, we have demonstrated and discussed some issues about the data-sets for image annotation. Firstly, we show how the simple propagation method CSD-Prop achieves fairly good results on the Corel set. It is argued that the Corel test set is a relatively easy set to be annotated when training on the Corel training set, because the Corel training and test sets contain many very globally similar images. Secondly, we have shown that the Corel test images can still be annotated well even when only 25% of the training information is used and it is argued that the training set contains redundant information.

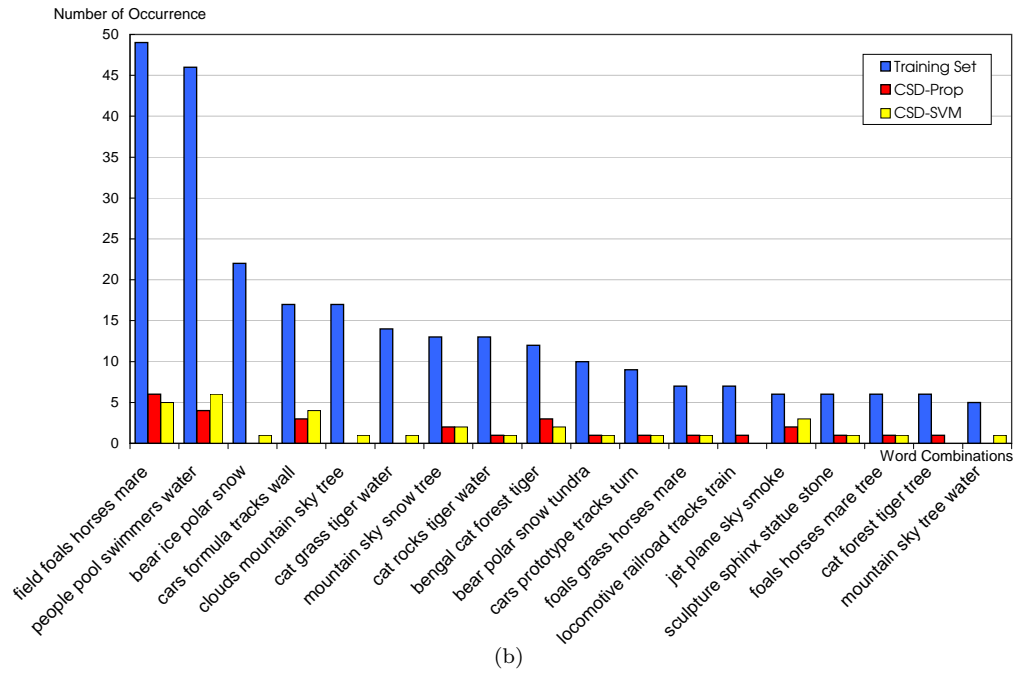
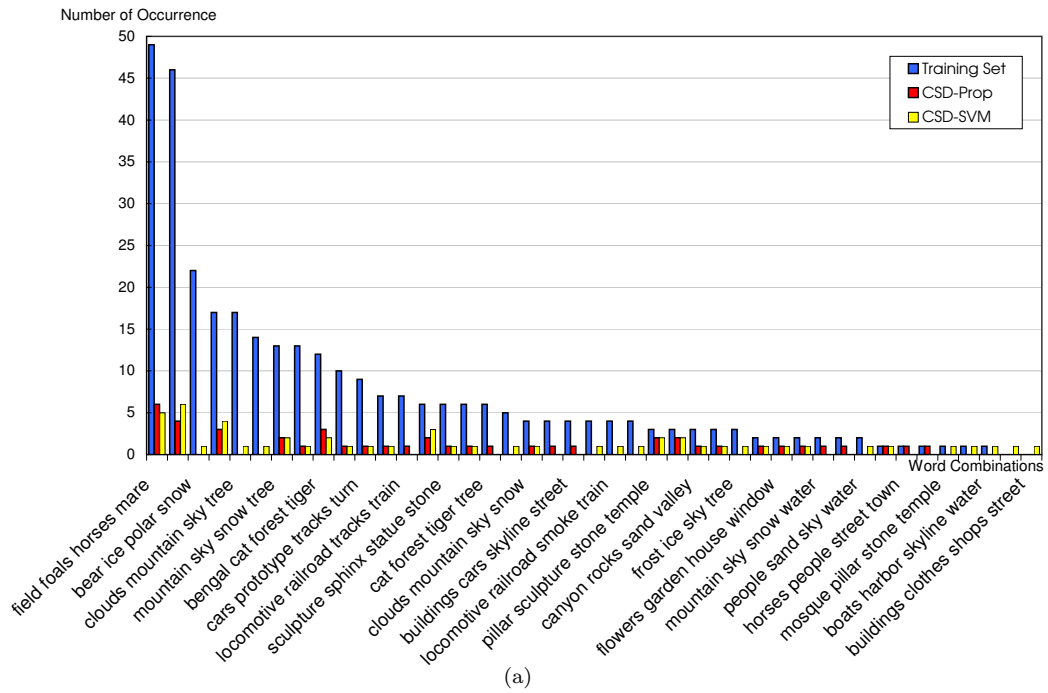


FIGURE 4.4: (a) Four word combinations that are correctly predicted by CSD-Prop and CSD-SVM, being ordered by the number of occurrences of each combination in the Corel training set. (b) Four word combinations that are correctly predicted by CSD-Prop and CSD-SVM, and with the number of occurrences greater than 5 in the Coral training set.

Models	Translation	CRM	MBRM	Mix-Hier	CSD-Prop	SvdCos	CSD-SVM
words with recall>0	49	107	122	137	130	102	127
Results on 49 best words							
Mean Per-word Recall	0.34	0.70	0.78	—	0.80	0.59	0.84
Mean Per-word Precision	0.20	0.59	0.74	—	0.58	0.51	0.74
Results on all 263 words							
Mean Per-word Recall	0.04	0.19	0.25	0.29	0.27	0.15	0.28
Mean Per-word Precision	0.06	0.16	0.24	0.23	0.20	0.15	0.25

TABLE 4.2: Comparison between CSD-Prop, SvdCos, CSD-SVD and some other state-of-the-art methods using the Corel images

Training Set	Corel(4500)			Yahoo(5260)		
Test Set	Corel(500)					
Models	CSD-Prop	SvdCos	CSD-SVM	CSD-Prop	SvdCos	CSD-SVM
Words with recall>0	107	100	94	46	58	59
Results on all 263 words						
Mean Per-word Recall	0.19	0.15	0.187	0.053	0.057	0.067
Mean Per-word Precision	0.14	0.11	0.153	0.038	0.040	0.053
Results on all 500 test images						
Cumulative Correct Annotations	577	349	767	102	123	118
Mean Per-image Recall	0.327	0.196	0.434	0.058	0.069	0.066
Mean Per-image Precision	0.231	0.140	0.306	0.040	0.049	0.047

TABLE 4.3: Comparison between the three methods on different training sets



	 1061	 119088
Ground-truth	clouds, sun, water, tree	buildings, clothes, shops, street
CSD-Prop	fountain, palace, light, reflection, palm	costume, street , village, buildings , people
CSD-SVM	sun , water , palm, tree , clouds	street , people, shops , clothes , buildings

FIGURE 4.5: Two combinations predicted by CSD-SVM that do not exist in the training set, words in bold being correct.

Chapter 5

Incorporating a Statistical Model with Salient Regions

Image auto-annotation, which automatically labels images with keywords, has been gaining more and more attentions in recent years. The advance of this technique could turn the traditional way of content based image retrieval (CBIR) which uses low-level image features (colour, shape, texture, etc.) as the query, into an approach that is more favorable to people, namely using descriptive words (semantics).

Previously, researchers have tended to use region-based image descriptors for image auto-annotation; Object-shaped regions generated by segmentation algorithms or uniform, usually rectangular, regions have been popular choices. Rectangular regions are a poor choice for image description because they are not robust to a variety of common image transformations, such as rotation. Current segmentation algorithms are not able to perfectly associate segmented regions to the actual objects that are being described. Undoubtedly, segmentation that is conducted by a fallible algorithm will have an adverse effect on the effectiveness of the auto-annotation algorithm.

This chapter is based on the publication by the author in (Tang et al., 2006), which presents an approach to image auto-annotation using a statistical model. However, unlike previous approaches this is achieved not by segmenting images but by using salient regions. It firstly introduces statistical models for image auto-annotation, in particular, the Cross-Media Relevance Model (CMRM) in details. Secondly, it proposes a way in which the salient region representation of images can be incorporated into the CMRM model. Finally, the experiment results are presented.

5.1 Statistical Models for Image Annotation

Statistical models try to reveal the association between visual features and keywords by estimating the joint probability distribution of regional image features and keywords, over a set of labelled training images. Given an unlabelled test image, the joint probability of its visual features and each keyword from the vocabulary can be calculated based on the association previously learnt. Some models attempt to annotate image regions (Duygulu et al., 2002; Barnard et al., 2003), whilst others annotate the whole image (Jeon et al., 2003; Lavrenko et al., 2003; Feng et al., 2004). The Cross-Media Relevance Model (CMRM) (Jeon et al., 2003), described briefly below, is in the latter class of models.

5.1.1 The Cross-Media Relevance Model (CMRM)

Following the derivation of Jeon *et al.* (Jeon et al., 2003), the CMRM model can be described as follows. Suppose there exists a training collection T , of labelled images, and a test collection Q , of unlabelled images.

Firstly, each training image is partitioned into shaped or uniform regions. Secondly, visual features, such as colour, shape or texture, are computed for each region. All of these regional features are clustered according to the similarity between them. These clusters, called ‘blobs’ (Barnard et al., 2003), can be viewed as *visual* words. Each image in the training set can thus be represented as a set of blobs, $B = \{b_1, \dots, b_n\}$, together with a set of annotation keywords, $W = \{w_1, \dots, w_m\}$. A joint probability distribution, $P(W, B)$, can then be constructed over the training set. In order to perform auto-annotation, the test images are also partitioned into regions, each of which is assigned to the blob that is closest to it. Thus, each test image can also be represented as a set of blobs $B = \{b_1, \dots, b_n\}$. The annotation process for an image is then a matter of finding the words that maximise the conditional probability $P(W|B) = P(W, B)/P(B)$. The joint probability $P(W, B)$ is computed as the joint expectation over the space of distributions $P(\cdot|J)$ defined by the training images $J \in T$. Specifically, given a test image $I \in Q$, whose blob representation is $B_I = \{b_{I_1}, \dots, b_{I_n}\}$, the following joint probability is computed for each word w from the vocabulary:

$$P(w, b_{I_1}, \dots, b_{I_n}) = \sum_{J \in T} P(J) P(w, b_{I_1}, \dots, b_{I_n} | J) . \quad (5.1)$$

The CMRM assumes that the events of observing w_i and b_{I_1}, \dots, b_{I_n} are mutually independent once a training image J is chosen. Therefore, equation (5.1) becomes:

$$P(w, b_{I_1}, \dots, b_{I_n}) = \sum_{J \in T} P(J) P(w|J) \prod_{i=1}^n P(b_{I_i}|J) . \quad (5.2)$$

5.2 Hybridising CMRM with a Saliency-based Image Representation

Most current statistical models annotate images by calculating the probability of keywords given the regional feature-vectors. This requires the images to be segmented, into object-shaped regions (Duygulu et al., 2002; Barnard et al., 2003; Jeon et al., 2003; Lavrenko et al., 2003) or uniform regions (Feng et al., 2004). However, segmentation algorithms are known to work imperfectly, and uniform regions are intuitively poor choices. That is to say, fallible segmentation potentially compromises the performance of auto-annotation. If the aim is to attach words to the entire image, instead of image regions, it is possibly beneficial to circumvent the segmentation stage.

Saliency-based image auto-annotation models (Hare and Lewis, 2005c) have shown some promise. They proposed a very simple method; annotations of the top M (1, 2 or 3) training images that best match the test image, in terms of visual similarity, are directly used as the annotations of the test image in question. The problem of this method is that it can not tell which of the annotations is the one most likely to be correct. In other words, it does not rank the keywords as statistical models do.

An alternative approach to auto-annotation, explored here, is to use statistical models with saliency, instead of segmentation. The use of a statistical model for annotation allows the keywords to be ranked by their probabilities. We have adopted the CMRM (Jeon et al., 2003) as the statistical model for our experiment and assume that a set of keywords is related to a set of visual terms created from salient regions. Specifically, instead of calculating the joint probability of keywords and image regions (blobs) (Jeon et al., 2003), we calculate the joint probability of keywords and a set of visual terms. As described in section 3.4.2, each training image, J , is represented by its saliency-based visual terms $S = \{s_1, \dots, s_n\}$ along with its annotations $W = \{w_1, \dots, w_n\}$. For each test image, I , the joint probability of each word from the vocabulary and its visual terms, $S_I = \{s_{I_1}, \dots, s_{I_n}\}$, is approximated as the expectation over the whole training set, as follows:

$$P(w, S_I) = \sum_{J \in T} P(J) P(w|J) \prod_{i=1}^n P(s_{I_i}|J) , \quad (5.3)$$

where, it is assumed that the events of observing w and s_{I_1}, \dots, s_{I_n} are mutually independent once a training image J is selected. $P(J)$ is treated uniformly as $1/N_T$, where N_T is the total number of training images. $P(w|J)$ and $P(s|J)$ are estimated by smoothed maximum likelihood, which is derived from (Jeon et al., 2003), as follows:

$$P(w|J) = (1 - \alpha) \frac{\#(w, J)}{|J|} + \alpha \frac{\#(w, T)}{|T|} , \quad (5.4)$$

$$P(s|J) = (1 - \beta) \frac{\#(s, J)}{|J|} + \beta \frac{\#(s, T)}{|T|} , \quad (5.5)$$

where, $\#(w, J)$ denotes the number of times word w occurs in the caption of J , and $\#(w, T)$ denotes the number of times word w occurs in all the captions of images in T . $\#(s, J)$ is the number of times saliency s occurs in J , and $\#(s, T)$ is that of the whole training set. $|J|$ is the aggregate count of all keywords and visual terms in J , and $|T|$ is that of the whole training set. α and β are smoothing parameters obtained by optimising system performance on a held-out portion of the training set.

In the end of the process, all of the words are ranked in the order of probability of being the correct annotation for the test image in question. The x top-ranking words are chosen as the annotations.

5.3 Results and Discussion

Direct comparisons between the saliency-based CMRM approach with the state-of-the-art methods (Duygulu et al., 2002; Barnard et al., 2003; Jeon et al., 2003; Lavrenko et al., 2003; Feng et al., 2004) on the Corel image set (Duygulu et al., 2002) are not available, because the Corel images at hand are all thumbnail sized. The small image size means most of the images have only between 10 and 20 salient regions which leads to a poor representation of the image content. However, we compare the saliency-based CMRM with the region-based CMRM, as detailed in (Jeon et al., 2003), on the University of Washington Ground Truth Image Database (University of Washington, 2004).

The Washington data-set contains 697 public-domain images, each of which has between 1 and 13 keywords indicating the image content. On average there are 4.8 keywords per image. The vocabulary size is 170. Detailed descriptions of the data-set can be found in Section 6.2.3.1.

As in the previous work, precision and recall, and the *normalised score* are used to measure the performance of our salient-based statistical auto-annotation method:

$$Recall = r/n \ , \quad (5.6)$$

$$Precision = r/(r + w) \ , \quad (5.7)$$

$$E_{NS}^{(model)} = \frac{r}{n} - \frac{w}{N - n} \ , \quad (5.8)$$

where, r is the number of correctly predicted words, n is the actual number of words in the test image, w is the number of wrongly predicted words, and N is the number of words in the vocabulary. See Section 2.3 and 6.2.3.2 for details.

5.3.1 Experimental Results of Auto-annotation by Saliency-based CMRM

We divided the data-set randomly into 3 parts, with 45% as the training set, 5% as the evaluation set and 50% as the test set. The evaluation set is used to estimate the smoothing parameters, α and β (Equation 5.4 and 5.5), for the CMRM model. Once the parameters are fixed, the training set and the evaluation set are merged to make a new training set, thus resulting in a training set (50%) and test set (50%) of the same size as that used in the work of Hare and Lewis (2005c). For the saliency-based CMRM, the number of visual terms was set to 3000 as used by Hare and Lewis (2005b). For the region-based CMRM, the optimum was found when the number of blobs was 300.

We compare our saliency-based CMRM method with the methods reported in (Hare and Lewis, 2005c), namely the LSI (Latent Semantic Indexing) model, the Vector Space model, random annotation and empirical frequency based annotation, and the region-based CMRM technique presented in (Jeon et al., 2003). The LSI model and the Vector Space Model have been introduced in Section 2.2.3.2. The frequency based approach chooses the keywords that occur most frequently in the entire training set as the annotations for any query image. Figure 5.1(a) shows the comparisons in terms of precision-recall curves. Error bars shown in the figure indicate the range of precision over 100 repeated runs, each of which used a random separation of the Washington set into training, test and evaluation sets. Figure 5.1(b) gives a zoomed in view of Figure 5.1(b). The curves for the saliency- and blob-based CMRM were generated by increasing the number of predicted words from 1 to 10. The curves for the LSI and the Vector Space Model were generated by increasing the number of images used for propagation (M), which are 1, 2 and 3. The Box-and-Whisker plot of the results are presented in Figure 5.2. The results are also summarised in Table 5.1, which gives an overall comparison. In order to give a more detailed comparison, for the Saliency-based CMRM and Region-based CMRM methods, we use the first 5 predicted labels of each test image to calculate the per-word precision and recall, as shown in Figure 5.3. Figure 5.4 shows some example images together with their true and predicted annotations.

The results show that auto-annotation using the hybrid CMRM with saliency works much better than by choosing words based on the frequency distribution. Saliency-based CMRM is also capable of predicting the probability of each word being the correct annotation for a test image. Words with higher probabilities are more likely to be correct. In the case that only one word for each test image is predicted, up to 80% of predictions are correct. Strictly speaking, this method performs slightly better than the LSI and Vector-Space models when approximately 5 words ($M = 1$) are predicted, but worse for 7 ($M = 2$) and 10 ($M = 3$) predicted keywords. However, considering the error bars, these three methods have very similar performances for 5, 7 and 10 words. This implies that the simple annotation propagation methods work almost as well as the statistical

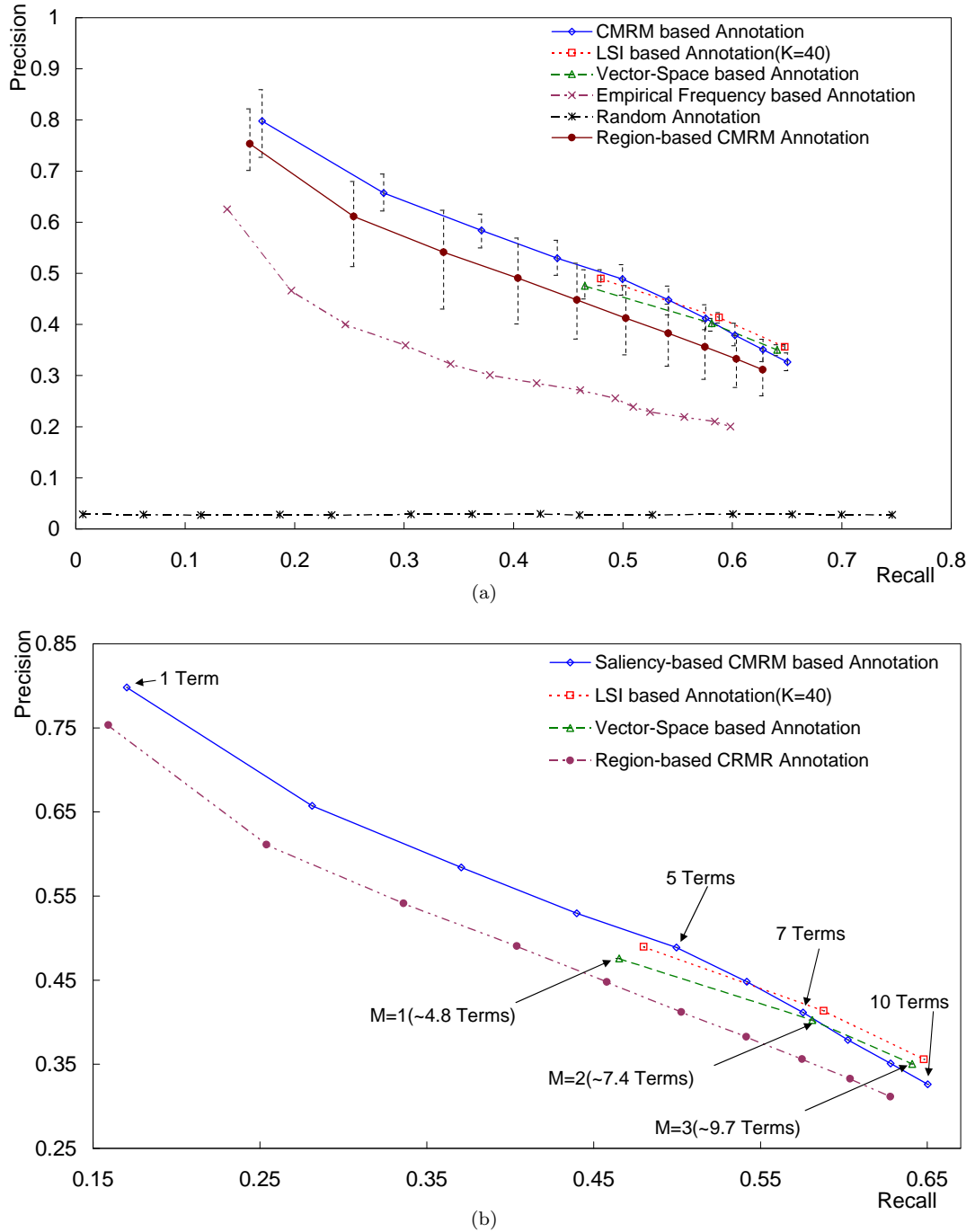


FIGURE 5.1: (a) Precision-Recall curves for several different auto-annotation methods. Error bars show range of precision over 100 repeated runs, each of which used a random separation of the Washington set into training, test and evaluation sets. (b) A zoomed in version of (a).

method. One possible reason, as argued by Monay and Gatica-Perez (Monay and Gatica-Perez, 2003), could be that propagating annotations can lead to good results when the data-set contains very similar images, which have almost the same set of annotations. This is the case for the Washington Dataset (University of Washington, 2004); If the right image is found, the exact annotations are also found. We can also see that on this

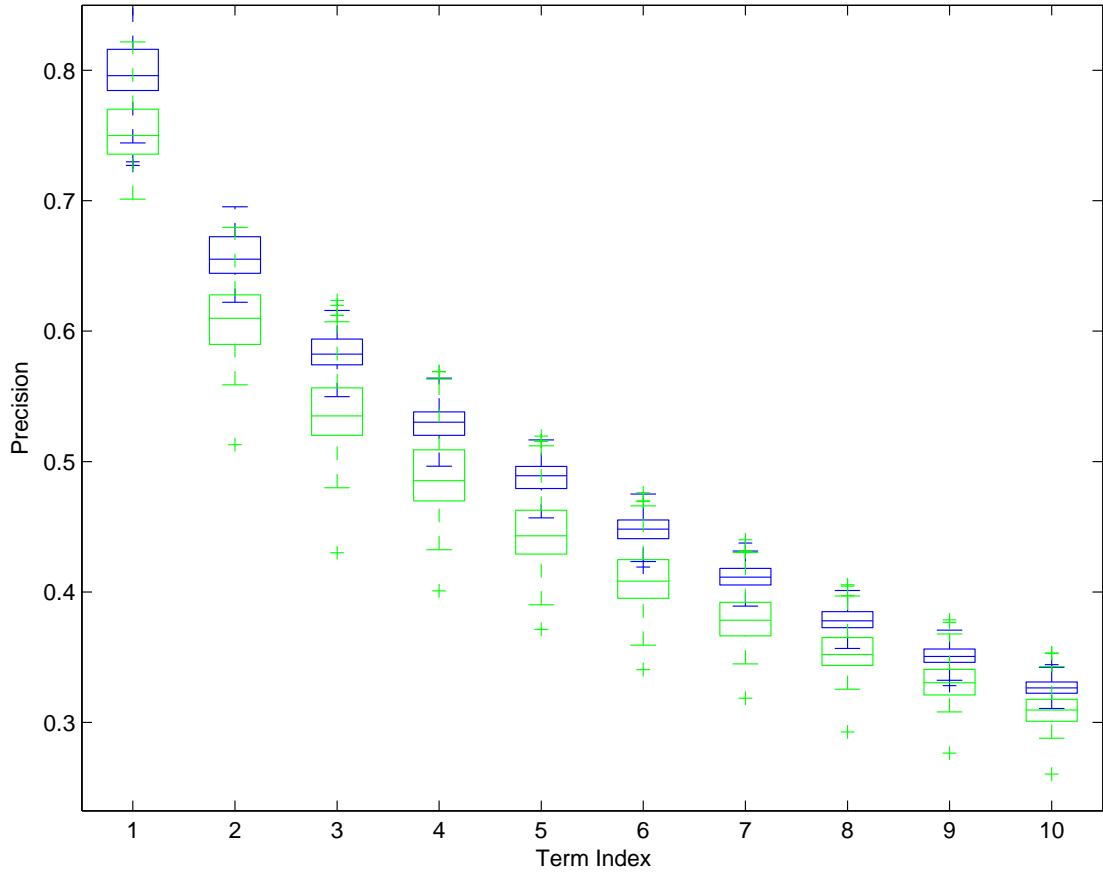


FIGURE 5.2: The Box-and-Whisker plot of the precisions of the saliency based CMRM approach (blue) and blob based CMRM approach (green). Results from 100 repeated runs are used. The horizontal axis represents the index of the predicted word, while the vertical axis represents the precisions.

data-set the saliency-based CMRM performs better than the region-based CMRM.

5.4 Summary

The chapter has proposed a new approach to image auto-annotation by using a statistical model coupled with an image description using salient regions. This approach avoids the image segmentation step taken by many previous auto-annotation techniques. The technique improves on the simple propagation-based annotation methods (LSI and Vector-Space) (Hare and Lewis, 2005c) in the sense that it is able to select individual words. It also improves on the CMRM model (Jeon et al., 2003) which uses general image regions/segments.

Method	Number of Words	Precision	Recall	E_{NS}
Saliency-based CMRM	1	0.800	0.170	0.169
	2	0.657	0.281	0.277
	3	0.584	0.371	0.363
	4	0.530	0.440	0.429
	5	0.489	0.500	0.484
	6	0.448	0.542	0.522
	7	0.412	0.576	0.551
	8	0.379	0.602	0.572
	9	0.351	0.628	0.593
	10	0.326	0.650	0.610
Region-based CMRM	1	0.753	0.159	0.158
	2	0.611	0.254	0.249
	3	0.541	0.336	0.328
	4	0.491	0.404	0.392
	5	0.448	0.458	0.441
	6	0.412	0.503	0.481
	7	0.383	0.541	0.515
	8	0.356	0.575	0.544
	9	0.333	0.604	0.567
	10	0.312	0.628	0.586
Vector-Space	~ 4.8	0.476	0.465	0.450
	~ 7.42	0.402	0.581	0.554
	~ 9.70	0.350	0.641	0.602
LSI(K=40)	~ 4.8	0.490	0.480	0.466
	~ 7.42	0.414	0.588	0.561
	~ 9.70	0.356	0.648	0.609

TABLE 5.1: Summary of Results

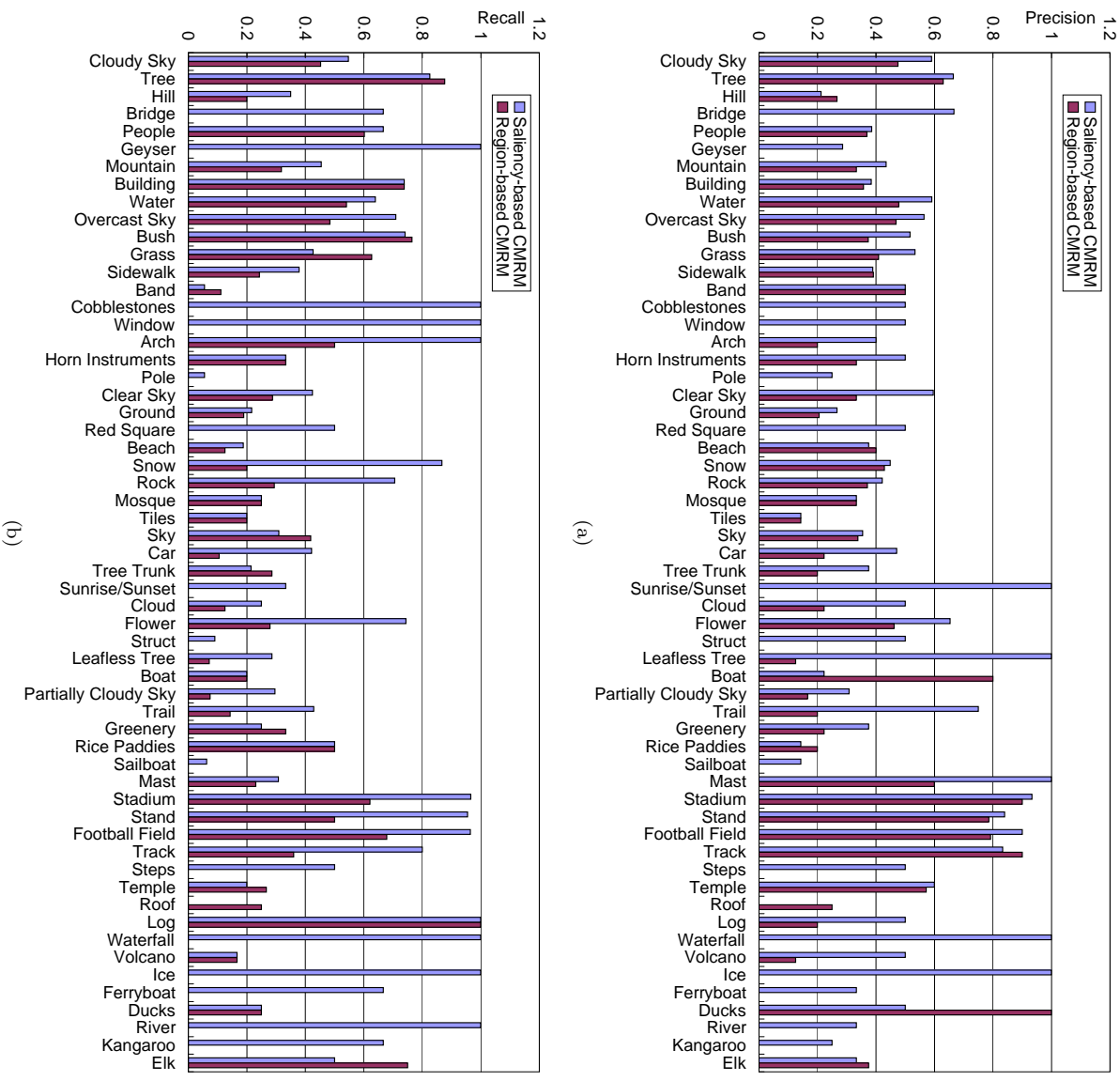


FIGURE 5.3: Per-word precision and recall of annotations predicted by Saliency-based CMRM and Region-based CMRM on the Washington set. (a) Per-word precision. (b) Per-word recall.




Images Methods			
True Annotations	Tree, Bush, Sidewalk	Temple, Sky	Flower, Bush, Tree, Sidewalk, Building
Empirical Annotations	Tree, Building, People, Bush, Grass	Tree, Building, People, Bush, Grass	Tree, Building, People, Bush, Grass
Vector-Space Annotations	Tree, Bush	Tree, Building, Grass, Sidewalk, Pole, People, Clear Sky	Flower, Bush, Tree, Building, Partially Cloudy Sky
LSI Annotations	Tree, Bush, Grass, Sidewalk	Steps, Wall	Flower, Bush, Tree, Ground
Region-based CMRM Annotations	Tree, Flower, Building, Bush, Overcast sky	Tree, Building, People, Clear sky, Cloudy sky	Tree, Building, Bush, Flower, People
Saliency-based CMRM Annotations	Tree, Cloudy sky, Bush, Overcast sky, Post	Clear sky, Rock, Snow, Tree, Building	Tree, Bush, Flower, Ground, Building

FIGURE 5.4: Example Annotations

Chapter 6

Non-negative Matrix Factorisation

In information retrieval, sub-space techniques are usually used to reveal the latent semantic structure of a data-set by projecting it to a low dimensional space. Non-negative matrix factorisation (NMF), which generates a non-negative representation of data through matrix decomposition, is one such technique. It is different from other similar techniques, such as singular vector decomposition (SVD), in its non-negativity constraints which lead to its parts-based representation characteristic. In this chapter, we present the novel use of NMF in two tasks; object class detection and automatic annotation of images. The contents of this chapter is based on the publication by the author in (Tang and Lewis, 2008).

6.1 Using Non-negative Matrix Factorization (NMF) to Discover Object Classes and Their Extent

The Vector Space Model (VSM), which represents a collection of documents by a term-document matrix, has been a major and popular model in information retrieval. A document can be text, image, or even video, while the observations made about it's content are referred to as terms. The term-document matrix is built in which each column represents a document, each row identifies a term and element values are the number of occurrences of a term in a document. For example, for a collection of images, each column of the matrix corresponds to an image and each item of the column indicates the number of times a certain visual term appears in the image. Visual terms have been chosen in many forms, for example 'blobs' (Pan et al., 2004), quantised salient regions (Hare and Lewis, 2005c), global RGB histogram (Hare et al., 2006), and even single pixels (Lee and Seung, 1999).

Usually the term-document matrices are high-dimensional and noisy, which makes it difficult to capture the underlying semantic structure. Dimensionality reduction techniques, e.g. principal component analysis (PCA), have been developed to reduce the dimensionality, filter noise, and discover the latent semantic structure. Recently, Lee and Seung (1999) proposed a new matrix decomposition technique called non-negative matrix factorisation (NMF). It is distinguished from PCA by its non-negativity constraints, which lead to its unique feature - parts-based representations of documents. They have shown that NMF is able to learn basis images that, for example for face images, correspond to face parts, such as mouth, nose and eyes. By contrast, PCA generates basis image, or eigenfaces (Turk and Pentland, 1991), which do not have an obvious visual interpretation, as shown in Figure 6.1. NMF has been applied for text document retrieval (Tsuge et al., 2001; Xu et al., 2003), image patch classification (Guillamet et al., 2002, 2003), and object recognition (Liu and Zheng, 2004).

In this section, we apply NMF to a collection of nature scene images, in order to discover the visually similar object or scene classes, without utilizing the captions attached with the images. Moreover, multiple segmentations are introduced to explore whether NMF is able to find more accurate segmentations, or more accurate extents of objects within images. The rest of the section is organised as follows. Firstly, we introduce the NMF technique, in comparison with PCA. Secondly, we present our approach to discovering the object classes from a set of images. Finally, we show the experimental results.

6.1.1 NMF vs. PCA

As we have mentioned, the Vector Space Model (VSM), or term-document matrix model, is a popular technique for information retrieval. Considering the quantity of information nowadays, it is usually the case that the matrix to be processed is very large. Similarity or distance measures on such high dimensional matrices are computationally expensive. On the other hand, it is difficult to find the real data structures, which are of most interest to researchers, in such massive data which are also contaminated by noise. As mentioned above, subspace techniques which project multi-dimensional data to a lower dimensional space have been developed. PCA and NMF are such techniques.

6.1.1.1 Principal Component Analysis (PCA)

PCA is a widely adopted technique from applied linear algebra for analyzing high dimensional data. The goal of PCA is to compute the most meaningful basis to re-express a data-set. It is hoped that the new basis will filter out the noise and reveal the underlying structure. Mathematically, it is an orthogonal linear transformation that transforms the data to a new coordinate system in which the variance of data is maximised on the first coordinate, the second greatest variance on the second coordinate, and so on. The

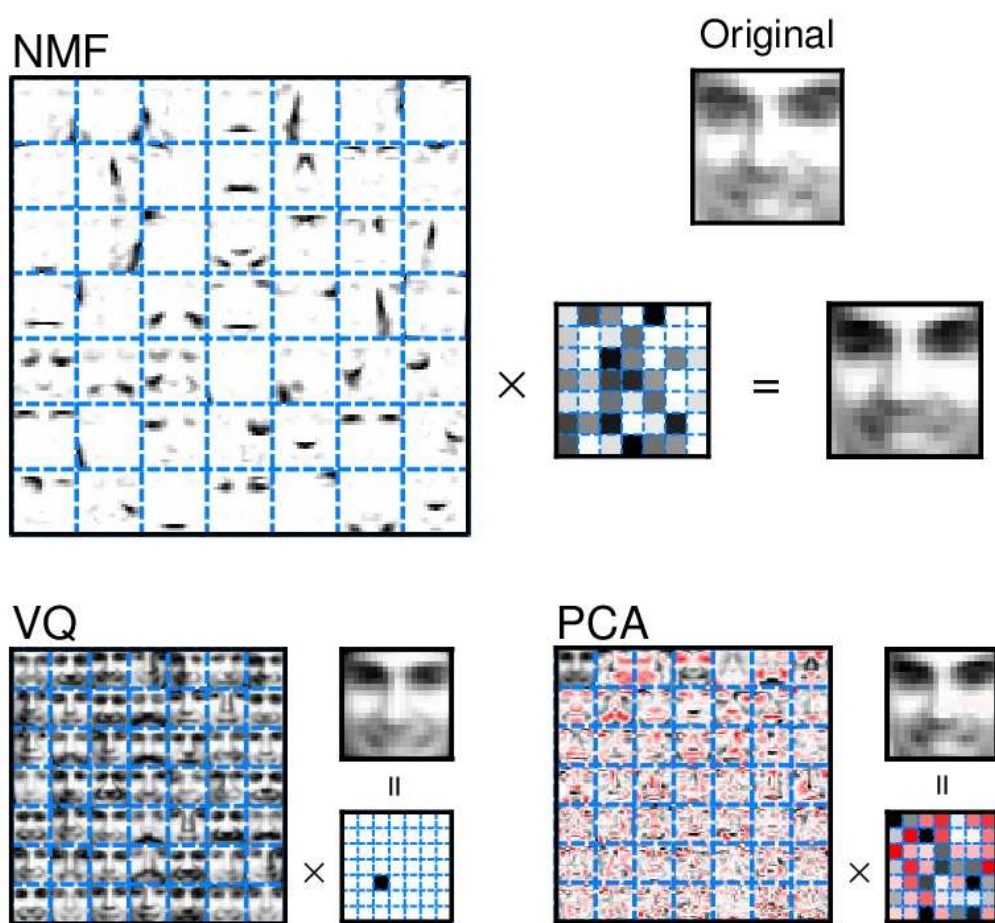


FIGURE 6.1: Parts-based representation of faces learnt by NMF and holistic representation learnt by PCA (Lee and Seung, 1999).

first coordinate is called the first principal component and the second is called the second principal component, and so on. Since the lower-order principal components capture the characteristics of data that contribute most to its variance, PCA can be used for dimensionality reduction by keeping the lower-order principal components and discarding the higher-order ones.

PCA can be solved by linear algebra techniques. Given a data-set of m samples/vectors, each of which is n -dimensional, it can be formed as a $n \times m$ matrix V . The data is pre-processed so that all the vectors have zero means, which can be done by subtracting the mean of all vectors. It can be proved (Jolliffe, 2002) that principal components can be found by eigenvector decomposition of the covariance matrix VV^T as follows

$$VV^T = EDE^T$$

where E is a matrix of eigenvectors of VV^T arranged as columns and D is a diagonal matrix. The principal components of V are the eigenvectors of VV^T , or the columns of E . In order to reduce dimensionality, V is projected onto a subspace spanned by the most important (lower-order) principal components, calculated as

$$V' = E_k^T V$$

where E_k^T contains the k eigenvectors corresponding to the k largest eigenvalues. As a result, similarity and distance measures can be conducted in this new space which is of lower dimensionality.

6.1.1.2 Non-negative Matrix Factorisation (NMF)

NMF is a technique to find a representation of non-negative data using non-negativity constraints. Under such constraints only additive, not subtractive, combinations are allowed, which leads to a parts-based representation of the original data. Given an $n \times m$ term-document matrix V (the same terminology as used for PCA) with $V_{ij} \geq 0$ and a pre-defined positive integer r , NMF finds two non-negative matrices $W \in R^{n \times r}$ and $H \in R^{r \times m}$ so that

$$V \approx WH$$

The rank r is generally chosen as smaller than n and m , for example $(n + m)r < nm$. NMF approximates each column of V by a linear combination of r column vectors in W . In other words, each column of W is regarded as a “basis” vector, while each column of H contains the weights needed for approximation. In order to estimate the factorisation matrices, an objective function needs to be defined. Lee and Seung (1999) defined one as

$$F = \sum_{i=1}^n \sum_{\mu=1}^m [V_{i\mu} \log(WH)_{i\mu} - (WH)_{i\mu}]$$

They also gave the following multiplicative update rules to minimize the difference between V and WH . Convergence is ensured (Lee and Seung, 2001).

$$\begin{aligned} W_{ia} &= W_{ia} \sum_{\mu} \frac{V_{i\mu}}{(WH)_{i\mu}} H_{a\mu}, \\ W_{ia} &= \frac{W_{ia}}{\sum_j W_{ja}}, \\ H_{a\mu} &= H_{a\mu} \sum_i W_{ia} \frac{V_{i\mu}}{(WH)_{i\mu}} \end{aligned}$$

However, the exact form of the objective function “is not as crucial as the non-negativity constraints for the success of the NMF algorithm in learning parts” (Lee and Seung, 1999). The objective function could be for example simply the Euclidean distance between V and WH as follows (Lee and Seung, 2001; Lin, 2005)

$$F = \|V - WH\| = \sum_{ij} (V_{ij} - (WH)_{ij})^2$$

6.1.1.3 Relation of NMF and PCA

Although both NMF and PCA are subspace techniques for dimensionality reduction, the bases they generate are different. PCA finds the directions of largest variance in data and constrains the principal components to be orthogonal to each other. It allows the values in the basis to be of arbitrary sign. Thus, when Lee and Seung (1999) apply PCA to a collection of human faces, the basis vectors, or eigenfaces (Turk and Pentland, 1991), generated by PCA do not have intuitive meaning. By contrast, for NMF, because there are no negative elements in W and H , only additive combinations are allowed. This is compatible with the intuitive notion of combining parts to form a whole. The “basis” vectors generated by NMF appear to be face parts such as mouths, noses, as shown in Figure 6.1. Each face image is approximated by a combination of the parts, using different weights.

6.1.2 NMF for Object Class Detection

Inspired by the work of Lee and Seung (1999) in which NMF was used to find the parts that form the whole faces, we explore its application to more general images, for example natural outdoor images. The idea is straightforward - we consider nature scene images as analogous to face images at the whole image level, then the objects (e.g. sky, water, tree, etc.) that constitute the nature scene images are analogous to the face parts (i.e. mouth, eye, nose, etc.) at the objects level. As the basis generated by NMF on the face images correspond to face parts, it is likely that the basis generated on outdoor images will correspond to natural objects or object parts.

The problem to be explored here can be formalised as follows. Given a collection of unannotated images, is it possible to learn the object classes simply from their appearances? An object class is a group of objects which may slightly differ from each other visually but correspond to the same semantic concept, e.g. the object class of ‘buildings’. We propose to answer the question in two main steps. Firstly, use NMF to find the bases which may correspond to objects. Secondly, we will try to use segmentation to segment out object regions and rank all the image segments by the distances to each basis object to see if the basis actually represent different object classes or object part classes.

Lee and Seung (1999) used the grey level pixel values of the face image to construct the term-document matrix. Each column is a face image and each element in the column corresponds to a pixel. Since the images used are 19×19 , it does not cause a problem when all the pixels are used. However, it can result in a very large matrix with general-purpose images which are often of high resolution. Resizing the images will lose a lot of information, and make them hardly recognisable at a resolution level as low as 19×19 . Therefore, another method of image representation is preferred. We choose again the quantised SIFT descriptors of salient regions, as used in many of our experiments (e.g. chapter 5 and 7.1). Let us briefly review the process here. Firstly, we select salient regions by using the method proposed by Lowe (2004). Secondly, Lowe’s SIFT descriptor is calculated for each salient region. Lastly, the k -means clustering algorithm is applied to the whole set of SIFT descriptors in order to quantise them into visual terms. As a result, each image can be represented by a vector which contains the number of occurrences of each visual term in the image. All the vectors are arranged as columns to form a matrix. Suppose the image collection is I_i ($i = 1, 2, \dots, m$, where m is the total number of images), mathematically we now have a $n \times m$ term-document matrix V , where V_{ij} is the occurrence of the i th visual term in image I_j , and n is the size of the visual vocabulary. Details of the process can be found in section 3.4.2.

Since all the elements in the term-document matrix are non-negative, we can now apply NMF to it. We adopted the projected gradient method for NMF that is developed by Lin (2005), because it converges faster than the popular multiplicative update approach (Lee and Seung, 1999). NMF decomposes the term-document matrix V into W and H where $V \approx WH$. If W is represented by its column vectors as $W = [W_1, W_2, \dots, W_r]$ (r is the number of basis vectors, or the dimensionality of the subspace), W_i is considered as a basis vector, or a conceptual part/object of the image collection. Each element of a basis vector indicates how many times a particular visual term appears in this conceptual object. In order to demonstrate if the basis vectors (i.e. W_i) actually correspond to object classes, the following approach is chosen. We use Normalized Cuts (Shi and Malik, 2000), which is an automatic image segmentation algorithm, to divide each image into regions. The visual terms within a specific region are used to form a vector representing the region. As a result, all the segments from the data-set can be represented by vectors I^t ($t = 1, 2, \dots, M$, where M is the total number of segments in

the data-set). For each basis vector, we rank all the image segments according to the distance, which is calculated as the cosine value of the angle between the two vectors, $\cos(W_i, I^t)$ ¹. The top ranked segments are examined to see if they represent an object class, which corresponds to the basis vector. So far, we have been ignoring the parameter r , the dimensionality of the subspace after NMF.

6.1.3 Experimental Results and Discussion

For comparison purposes, we choose the same data-set as used by Russell et al. (2006) for our experiments. The data-set is a subset of a large image database named LabelMe (Russell et al., 2005). The subset used in this work has 1554 images which are returned by the query words “cars”, “trees” and “buildings” from the LabelMe set. The images also contain many other additional objects. Most of the images have a resolution of 640×480 . The size of the visual vocabulary is set to 2000, which results in a term-document of 2000×1554 . Finding the optimal value of r is a difficult problem in itself. In this work, we only set the value empirically to 35. In terms of segmentation, we followed the setting of the work by Russell et al. (2006). Specifically, to produce multiple segmentations, we varied two parameters of Normalised Cuts: the number of segments K and the size of the input image. K is set to 3, 5, 7, 9, 11, 13 and the segmentation algorithm is applied at 2 image scales: 50- and 100-pixels across. This results in 12 different levels of segmentation per image. For each basis vector, segments from all segmentation levels are ranked according to their distances to the basis vector. Figure 6.2 shows montages of segments for the object classes found by NMF, each corresponding to a basis vector of W . Each of the depicted segments comes from a different image to avoid showing multiple segments of the same object. Making a qualitative assessment; it can be seen, NMF manages to find some object classes (e.g. “trees”, “sky”, “buildings”, etc.) fairly well. However, it was not successful on “cars” in our experiments.

We also conducted an evaluation on the effect of using multiple segmentations. Segmentation accuracy, which was used by Russell et al. (2006), is calculated on the top ranked LabelMe segments for each object class. The top twenty returned images for four object classes are tested: “buildings”, “cars”, “roads” and “sky”. They are compared with the ground truth object segmentation that was generated manually. Suppose R and GT denote the set of pixels in the retrieved object segment and the ground truth segmentations of the object. The accuracy score, ρ , measures how correct the area segmented by the retrieved object segment is. It is calculated as the ratio of the intersection of GT and R to the union of GT and R , as follows

$$\rho = \frac{GT \cap R}{GT \cup R}$$

¹For vectors V_1 and V_2 , $\cos(V_1, V_2) = \frac{V_1 \cdot V_2}{|V_1||V_2|}$

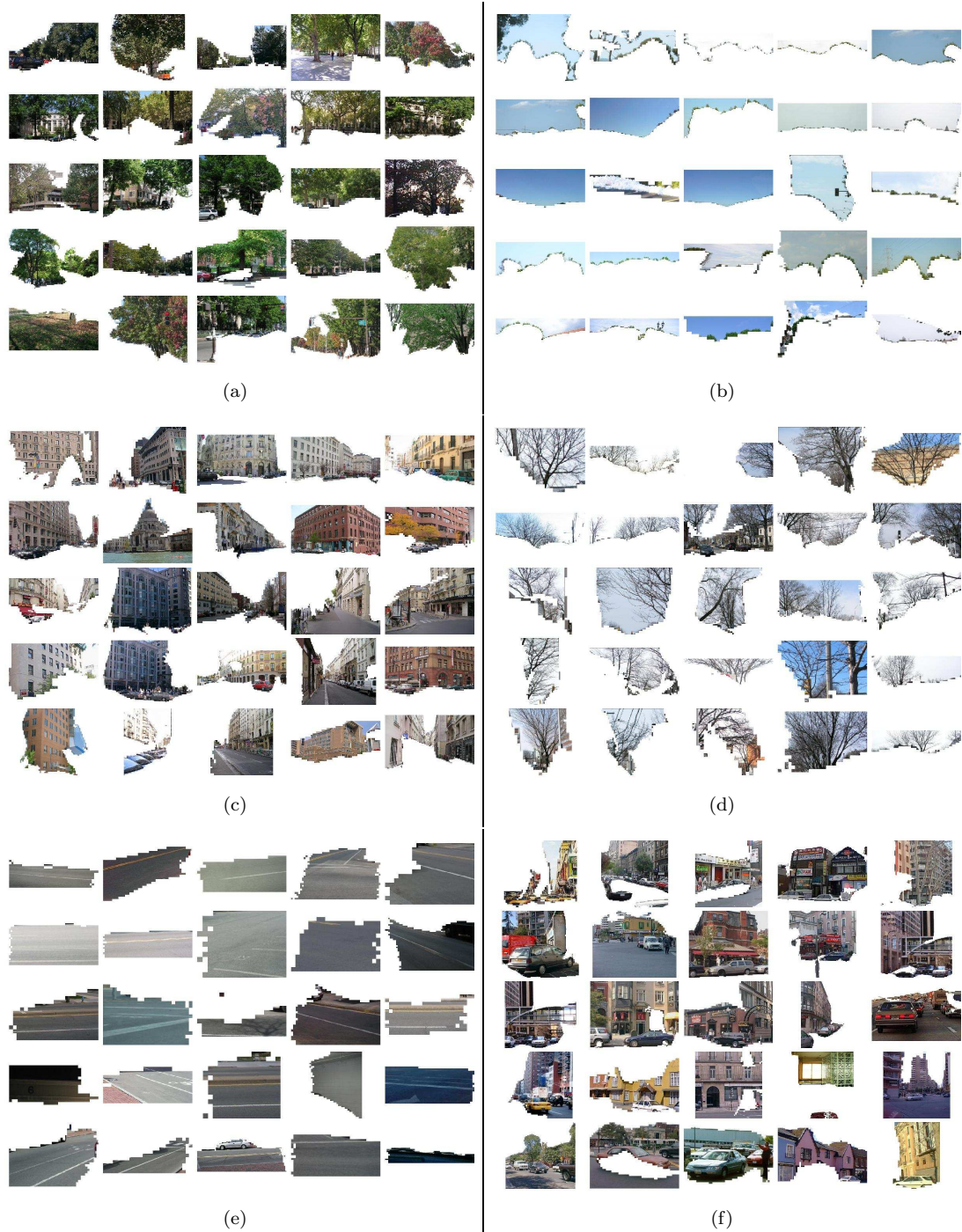


FIGURE 6.2: Top segments for 6 (out of 35) object classes discovered in the LabelMe data-set. Note how the segments, learned from a collection of unlabeled images, correspond to trees (a), sky (b), buildings (c), leafless trees (d), roads (e). However, for the last group of segments (f), it is not obvious which class of objects it corresponds to.

We consider it as the class of cars in our evaluations.

Methods	buildings	cars	roads	sky
Our Methods				
Multi Seg NMF	0.69	0.11	0.39	0.83
Sing Seg NMF	0.50	0.09	0.47	0.67
Russell et al.'s Methods				
Multi Seg LDA	0.53	0.21	0.41	0.77
Multi Seg pLSA	0.59	0.09	0.16	0.77
Sing Seg LDA	0.55	0.29	0.32	0.65

TABLE 6.1: Segmentation accuracy (ρ) of the top 20 segments returned by NMF on four object classes from the LabelMe dataset. It is compared with the results from Russell et al. (2006) on the same data-set.

If more than one ground truth segment intersects with R , we choose the one achieving the highest score. The accuracy score is averaged over the top 20 returned object segments for the four classes. The results are shown in Table 6.1. The table also includes the results of using single segmentation, for comparison with multiple segmentations. As can be seen, NMF performs as well as, if not better than, the methods proposed by Russell et al. (2006). In particular, NMF with multiple segmentations outperforms LDA (Latent Dirichlet Allocation) with multiple segmentations by 0.16 on “buildings” and 0.06 on “sky”, although it is worse by 0.02 and 0.10 on “roads” and “cars” respectively.

It is interesting to note the difference between our approach and that of Russell et al. (2006). Although both methods use quantised descriptors of salient regions as visual terms, we treat each entire image as a document, while Russell et al. treat each image segment as a document. We build a term-document matrix and then rely on NMF to find the basis vectors, or underlying “topics” as called by Russell et al. They apply statistical models to the whole set of image segments to find the “topics”. Therefore, the data which needs to be processed is significantly less in our approach.

6.2 Auto-annotation via Semantic Propagation in Sub-space

Latent semantic indexing (LSI) was proposed by Deerwester et al. (1990) to perform document clustering. They demonstrated that it is possible to reveal the implicit higher-order structure in the association of terms with documents, by projecting the term by document matrix into a sub-space through the singular value decomposition (SVD). Hare and Lewis (2005c) used this technique (SVD for LSI) for image annotation using semantic propagation. The premise of their approach is based on the intuition that visually similar images often have similar meaning or semantics. NMF as another matrix factorisation technique can be used as an alternative to SVD in order to project high dimensional data to a low dimensional sub-space, in which the semantics of data is expected to be more explicit. In this section, we will examine the use of NMF for image auto-annotation via semantic propagation.

6.2.1 NMF as an alternative to SVD

6.2.1.1 SVD

SVD is a popular technique used in the information retrieval community. It decomposes a $m \times n$ matrix A into the product of a $m \times r$ matrix T , a $r \times r$ matrix S , and a $r \times n$ matrix D :

$$A = TSD^T \quad (6.1)$$

such that $TT^T = DD^T = D^TD = I$, where I is the identity matrix. Figure 6.3 depicts a graphical representation of SVD. S is a diagonal matrix in which diagonal elements are called singular values of matrix A , in monotonically decreasing order. It is claimed that the k largest singular values together with the corresponding left and right eigenvectors encode the most important information of A (Deerwester et al., 1990). Therefore, matrix A is usually approximated by A^* (i.e. $A \approx A^*$), which is thought to contain less noise or be noise-free:

$$A^* = T_k S_k D_k^T \quad (6.2)$$

Suppose A is a term document matrix, each column of which corresponds to a document, we further define matrix T^* and D^* to be

$$\begin{aligned} T^* &= T_k \\ D^* &= S_k D_k^T \end{aligned} \quad (6.3)$$

where T^* is regarded as the term matrix and D^* is the document matrix (note that we could have equally chosen $T^* = T_k S_k^{1/2}$ and $D^* = S_k^{1/2} D_k^T$). The decomposition $A \approx T^* D^*$ can be interpreted as follows. Each column of T^* captures a basis of the latent semantics of the document corpus, while each element of a column in D^* indicates the index of the corresponding document on each basis. However, because of the negative values in some of the dimensions generated by SVD, the above interpretation becomes less meaningful.

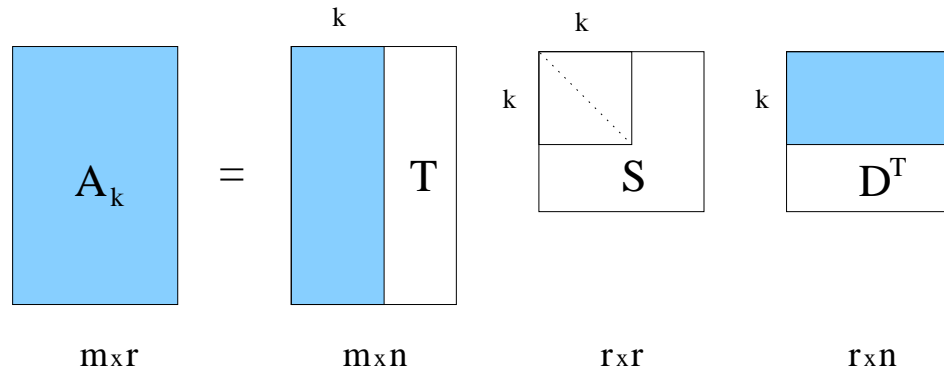


FIGURE 6.3: Graphical representation of SVD.

6.2.1.2 NMF for Sub-space Projection

Given a document corpus, or a collection of images, we assume that it consists of k topics. Note that we use the same terminology " k " as used in SVD where it refers to the k largest singular values. Each document in the corpus either completely belongs to a particular topic, or is partially related to several topics. Ideally, if the documents can be projected into a k dimensional semantic space in which each axis corresponds to one of the k topics, the semantic structure of the data-set will be more explicit. In other words, each document can be represented by a linear combination of the k topics. Because it is more natural to consider each document as an additive instead of subtractive mixture of different topics, the coefficients of the linear combination should be all non-negative. Moreover, it is usually the case that topics of a corpus are not completely independent of each other. Overlaps may exist among the topics. Therefore, the axes of the semantic space that capture the topics are not necessarily orthogonal, which is the case for the sub-space generated by SVD.

NMF is theoretically superior to SVD for the following reasons (see Figure 6.4). First, overlaps exist among topics. The orthogonal restriction by SVD makes the derived latent semantic directions less likely to correspond to each topic, while NMF can still find them. Second, NMF decomposes the matrix in such a way that each document can be considered as a additive combination of topics, which makes more sense in the image domain. Third, for a particular document, the coefficients of the linear combination in NMF give direct indications of to what extent this document is belonging to each of the topics. In contrast, SVD can not give this benefit because those negative values do not have intuitive interpretations.

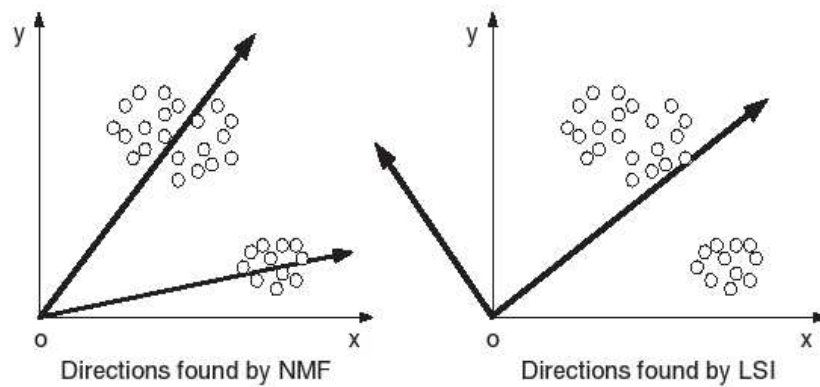


FIGURE 6.4: Illustration of the difference between NMF and SVD (Xu et al., 2003).

Based on the above theory, we propose to utilize NMF to find the latent semantic structure for a collection of images, and then use the semantic propagation method to annotate images automatically. Before the details of the approach are presented, a

modified versions of NMF that is used in our experiments will be discussed, namely the NMF with sparseness constraints.

NMF with sparseness constraints Hoyer (2004) noticed one of the most useful properties of NMF is that it generates a sparse representation of data. Much of the data is encoded in such a representation using only a few ‘active’ components. This notion is in line with the initial interpretation of NMF that parts are combined to build a whole. It is argued that in some applications, NMF does not result in parts-based representations because the decomposed matrices are not ‘sparse’ enough. Therefore, sparseness constraints are applied to the objective function, in order to achieve a pre-defined level of sparseness of the decomposition. They proved that in their experiment the parts-based representation of data can be enhanced through this approach.

“The concept of ‘sparse coding’ refers to a representational scheme where only a few units (out of a large population) are effectively used to represent typical data vectors” (Hoyer, 2004). In other words, most units take values that are close to zero and only a few take large non-zero values. Hoyer defines the sparseness of a vector x as follows, which is based on the relationship between L_1 norm and L_2 norm:

$$sparseness(x) = \frac{\sqrt{n} - (\sum |x_i|) / \sqrt{\sum x_i^2}}{\sqrt{n} - 1} \quad (6.4)$$

where n is the dimensionality of x . This function evaluates the sparseness of a vector to a value within the range of $[0,1]$. The sparseness equals 1 if and only if x contains only one single non-zero component; it equals 0 if and only if all components of x are equal. Figure 6.5 illustrates some examples of this sparseness measure.

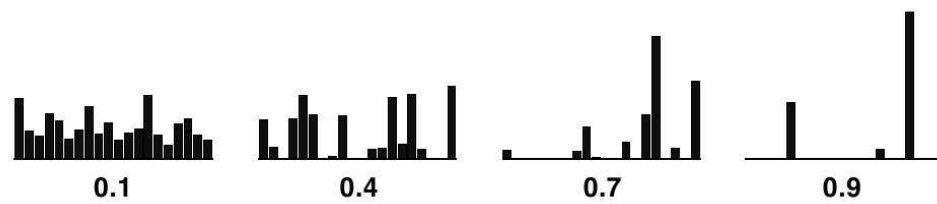


FIGURE 6.5: Illustrations of 4 different degrees of sparseness, 0.1, 0.4, 0.7, 0.9. The height of each bar denotes the value of one element of the the vector (Hoyer, 2004).

NMF with sparseness constraints is then defined as follows. Given a non-negative matrix V , find the non-negative matrices W and H such that

$$E(W, H) = \|V - WH\|^2 \quad (6.5)$$

is minimized, under optional constraints

$$\begin{aligned} sparseness(w_i) &= S_w, \forall i, \\ sparseness(h_i) &= S_h, \forall i, \end{aligned} \tag{6.6}$$

where w_i is the i th column of W and h_i is the i th row of H . S_w and S_h are the desired sparsenesses of W and H respectively, and are set by the user.

6.2.2 Using NMF and Semantic Propagation for Auto-annotation

Semantic propagation is perhaps the simplest automatic image annotation method. The basic idea is intuitive; images that have similar visual appearance should have similar semantics. For a given new un-annotated image, a CBIR-like process is carried out first in order to rank the training images which are already annotated. Then, labels are chosen from the top (most similar) training images to annotate the new image. Therefore, most of the traditional CBIR techniques can be directly transferred to image auto-annotation applications in the manner described above. For example, Hare and Lewis (2005c) search for visually similar images in the semantic space that is generated by applying SVD to the term by document matrix of an image collection, and then propagate the labels from the top ranked images (1, 2 and 3 respectively) to a new query image.

We choose the same approach as that of Hare and Lewis (2005c), except that NMF is used to find the latent semantic topics. The whole process is conducted as follows.

1. The saliency based visual term representation (see Section 3.4.2) of the training images are generated. The same image representation is calculated on the query images, except that the cluster centroids of salient descriptors found on the training set are used.
2. All the training images which are represented by vectors of visual terms are put together to form a term by document matrix V . Each column of V corresponds to an image, while each element of a column indicates the number of occurrences of a particular visual term.
3. NMF is applied on V to generate W and H such that $V \approx WH$. Each column of W is considered as a topic, while each column of H contains the coefficients of the linear combination of the corresponding topics.
4. Each query image q is projected into the semantic space spanned by W . Because we assume that the query image shares the same latent semantic structure as the training set, equation $q = Wh_q$ stands, where h_q is the new coordinates of q . h_q can be calculated as $h_q = W^{-1}q$.

5. Training images are ranked according to their distances to the query image in the space of W . In other words, we compare each column of H with h_q . Cosine distance of vectors is used in this work.
6. Labels of the top M training images are propagated to the new image as its predicted labels.

6.2.3 Experimental results

6.2.3.1 The Washington Image Data-set

For comparison, the same data-set as used by Hare and Lewis (2005c) is chosen for experiments, namely the University of Washington Ground Truth Data-set (University of Washington, 2004). The Washington data-set contains 697 public-domain images, each of which has been semantically annotated between 1 and 13 keywords. For example, an image may have several labels describing its content, such as “trees”, “buildings”, “sky”, etc. On average there are 4.8 keywords per image. After the original keyword labels were processed by correcting mistakes and merging plurals into singular forms (Hare and Lewis, 2005c), e.g. “trees” became “tree”, the vocabulary consisted of 170 keywords. The frequency distribution of keywords across the data-set is shown in Figure 6.7. Figure 6.6 shows some sample images and the corresponding annotations from the data-set.

6.2.3.2 Performance Evaluation

For each test image, precision and recall, as well as the *normalised score* proposed by Barnard *et al* (Barnard et al., 2003), are calculated for performance evaluation. Each kind of metric is averaged over the entire test set to get a mean value. The definitions of these metrics are as follows.

$$Recall = r/n \quad , \quad (6.7)$$

$$Precision = r/(r + w) \quad , \quad (6.8)$$

$$E_{NS}^{(model)} = \frac{r}{n} - \frac{w}{N - n} \quad , \quad (6.9)$$

where, r is the number of correctly predicted words, n is the actual number of words in the test image, w is the number of wrongly predicted words, and N is the number of words in the vocabulary. Details of the above metrics can be found in Section 2.3.

6.2.3.3 Experiment Settings

We compare the results of experimentations with three different settings of experiment on sub-space techniques for semantic propagation based image auto-annotation, i.e. classic



Tree, Bush, Grass, Sidewalk



Tree, Beach, Ocean, Sky, Cliffs



Clear Sky, Building, Ground, People, Red Square



Overcast Sky, House, Car, People, Struct, Flowers



Tree, Grass, Sidewalk, People, Clear Sky



Cloudy Sky, Bridge, Water, Tree



Stadium, Stand, People, Football Field, Number, People, Tree, Track, Line



Volcano, Sky, Cloud



Clay Houses, Sky, Minaret



Tree Trunk, Log, Greenery, Ground, Elk



Partially Cloudy Sky, Tree, Beach, Sailboat, Mast, Water, Shadows



Swan Boat, Lake, Tree

FIGURE 6.6: Sample images and their annotations from the Washington Ground Truth Image Database.

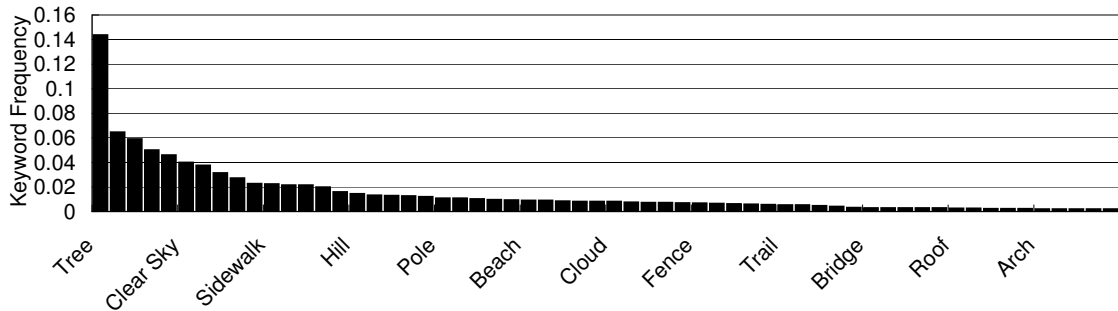


FIGURE 6.7: Plot of empirical keyword distribution in the Washington data-set

NMF (denoted as CNMF), NMF with sparseness constraints (denoted as NMFsc) and SVD. The results of the SVD based approach were taken directly from the work of Hare and Lewis (2005c) for comparison purposes. The projected gradients based method¹ developed by Lin (2005) was used for the classic NMF. As for NMF with sparseness constraints, the algorithm² developed by Hoyer (2004) was adopted.

Additional parameters need to be set when using the sparseness constrained version of NMF, namely the degree of sparseness of W and H . The constraints can be placed on W , or H , or both, depending on the particular problem to be solved. Hoyer (2004) described an example in which a doctor tries to analyze disease patterns. It was assumed that most diseases are rare (hence sparse), and present in a small number of patients. However, each disease can cause many symptoms. Therefore, given a matrix in which each column denotes an individual patient and each row denotes a symptom, it might be better to place sparseness constraints on the “coefficients” (rows in H) but not the “basis vectors” (columns in W). Based on empirical analysis of the Washington images set, we chose to constrain W but not H for two reasons. Firstly, as the number of visual terms was set to 3000 but on average each image generated only several thousand salient regions, it is unlikely that an object or object part from an image contains a variety of different visual terms. In other words, the “basis vectors” in W tend to be sparse. Secondly, many objects/keywords exist in a large number of images in the data-set. For example, 484 of the images contain “tree”, and 218 and 199 of them have “building” and “people”. These keywords affect a big portion of the data-set. It is more appropriate to unconstrain H . Our experiments also confirmed this hypotheses; the results of experiments using constrained W and unconstrained H were much better than using unconstrained W and constrained H , or when both were constrained. When both W and H are unconstrained, it becomes the classic NMF, the results of which are presented in the following.

¹Code available at: <http://www.csie.ntu.edu.tw/~cjlin/nmf/index.html>

²Code available at: <http://www.cs.helsinki.fi/u/phoyer/software.html>

6.2.3.4 Results

We repeated the experiments of CNMF and NMFsc for image auto-annotation 100 times on different training and test sets. For each run, a randomly selected 50:50 mix of images from the Washington data-set were used to build a set of training images and a set of test images. Precision, recall and normalised score (E_{ns}) were calculated at different values of $M(1, 2, 3)$, which represents the number of top images chosen for propagation. The average results from the 100 runs are used in the following. The number of visual terms was set to 3000 and the term by document matrix was not weighted.

The dimensionality of the sub-space generated by NMF, or the value of r in $V \approx W_{n \times r} H_{r \times m}$, is a number pre-defined by users. Theoretically, it should relate to the class number of object or object parts in the data-set. However, at this time, finding the optimal value of r is still a difficult and unsolved problem. In our experiments, we varied its value from 2 to 200 with a fixed step of 2. Besides, for NMFsc, the results were calculated at different sparseness degrees of W , i.e. 0.5, 0.6, 0.7, 0.8 and 0.9.

In order to choose the optimal sub-space dimensionality and degree of sparseness for NMFsc, we use normalised score as a single value indicator. Figure 6.8 depicts the values of E_{ns} at different settings of dimensionality (r) for different degrees of sparseness of W . For each test image, the closest training image was chosen for propagation, i.e. $M = 1$. The horizontal axis represents the value of r , and the vertical axis represents the value of normalised score E_{ns} . Each degree of sparseness generated one curve in the chart, denoted by different colours. Figure 6.9 and 6.10 show the results of using 2 and 3 top images for propagation respectively, i.e. $M = 2, 3$. As can be seen from the figures, E_{ns} achieves the highest value when the sparseness is 0.8 (the green curve) and r is around 100. We have also calculated the results for CNMF and depicted this in Figure 6.11. The best performance is found at $r = 40$, as shown in the chart. The above mentioned values of parameters are chosen for comparisons with SVD.

The results in terms of precision, recall and normalised score are summarised in Table 6.2, along with the results of the methods proposed by Hare and Lewis (2005c), namely the vector space and LSI (based on SVD) model. The results of each method are also plotted into a precision-by-recall chart, Figure 6.12, for a better view of the comparison. As can be seen, the annotation results of NMF with sparseness constraints are better than that of the classic NMF. Besides, NMFsc achieved similar results as SVD when $M = 1$, and slightly better when $M = 2$ and 3. Some samples of annotation results are shown in Figure 6.13.

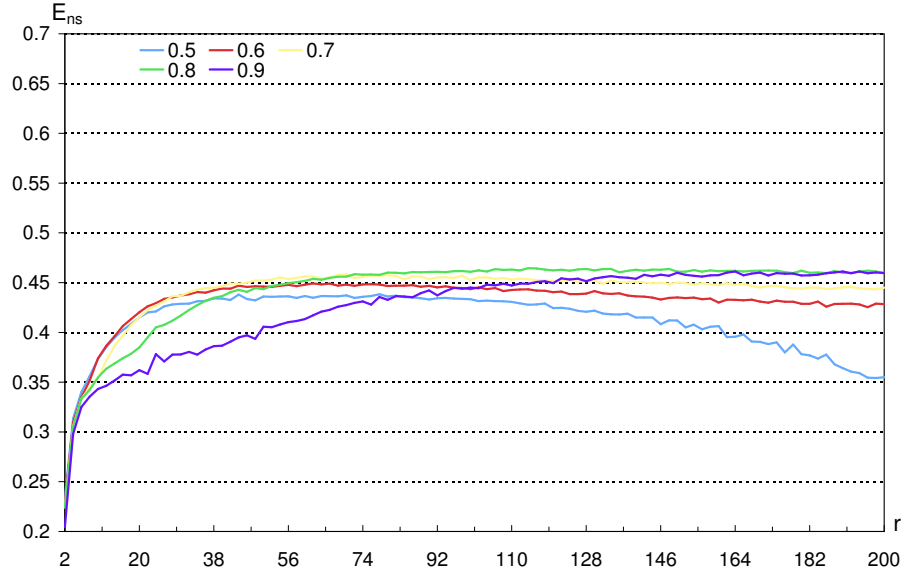


FIGURE 6.8: The normalised score (E_{ns}) of applying NMFsc for image auto-annotation. The closest training image ($M = 1$) is used for propagation. The degree of sparseness is varied from 0.5 to 0.9.

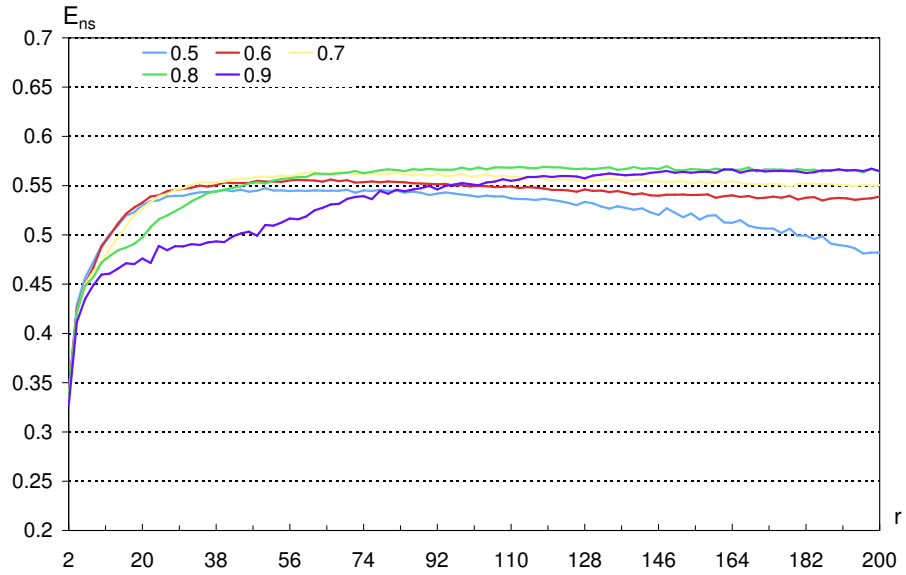


FIGURE 6.9: The normalised score (E_{ns}) of applying NMFsc for image auto-annotation. The top 2 closest training images ($M = 2$) is used for propagation. The degree of sparseness is varied from 0.5 to 0.9.

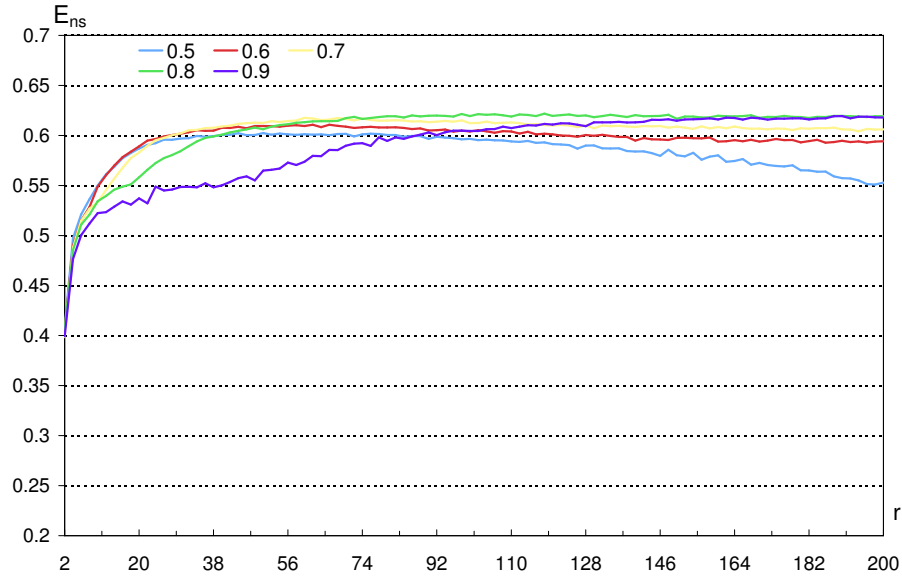


FIGURE 6.10: The normalised score (E_{ns}) of applying NMFsc for image auto-annotation. The top 3 closest training images ($M = 3$) is used for propagation. The degree of sparseness is varied from 0.5 to 0.9.

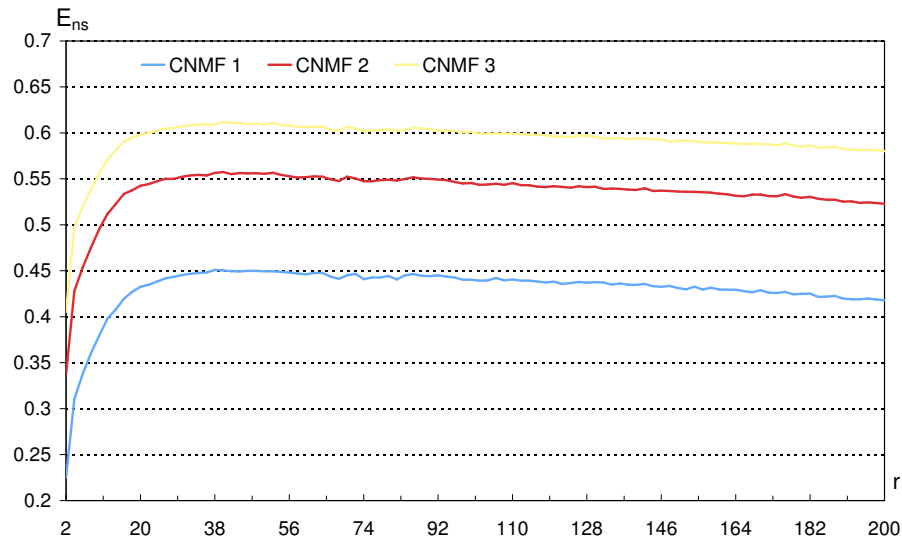


FIGURE 6.11: The normalised score (E_{ns}) of applying CNMF for image auto-annotation. The curve “CNMF 1” represents the results when $M = 1$. “CNMF 2” and “CNMF 3” represent the case when $M = 2$ and $M = 3$.

6.3 Summary

This chapter has investigated the use of the non-negative matrix factorisation (NMF) technique for object detection and image auto-annotation. In the first part, we have demonstrated an approach that utilizes the parts-based representation feature of NMF for object class or topic detection among a set of images without any labels. NMF is applied to the term-document matrix of an image data-set. The basis vectors in the decomposed matrix are considered as conceptual objects, each of which corresponds to an object class, or topic. It has also been shown that NMF manages to find more accurate

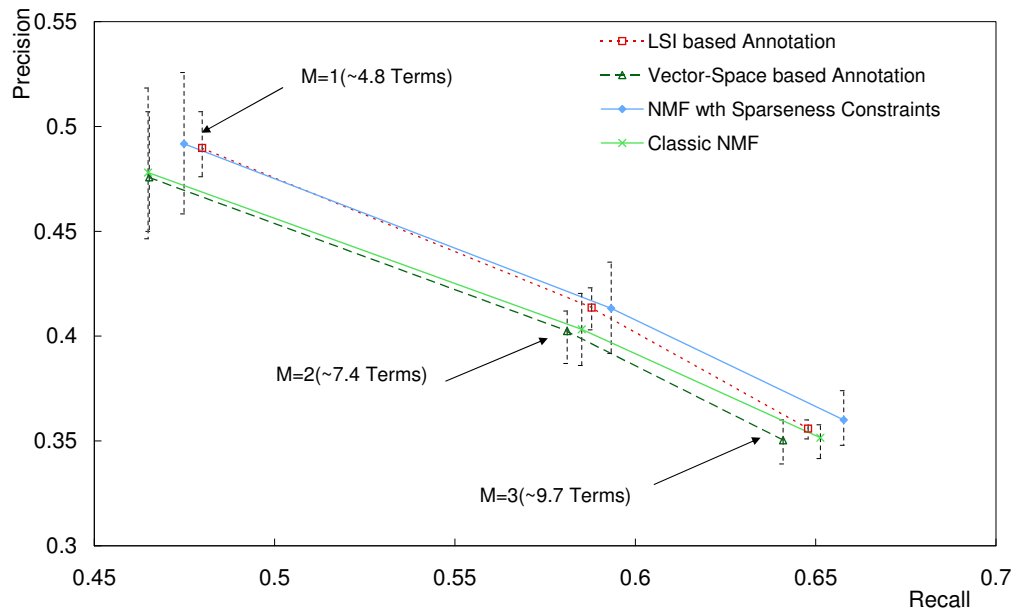


FIGURE 6.12: Precision-Recall curves for several different semantic propagation-based image auto-annotation methods. Error bars show range of precision over 100 repeated runs, each of which used a random separation of the Washington set into training and test sets.



Images Methods			
GT Labels	Clear Sky, Building, Tree, Leafless Tree, Tree Trunk, Grass, Street, Tree	Stadium, Stand, People, Football Field, Band, Post, Track, Banner	Cloudy Sky, Hill, Tree, Building, Water
CNMF	Tree, Grass, Pole, Building, People, Clear Sky, Water, Partially Cloudy Sky, Sidewalk, Elk	Tree, Stadium, Stand, Football Field, People, Band, Post, Track, Banner, Sky, Lake	Tree, Cloudy Sky, Boat, Water, Mountain, Dock, Powerboat, Mast, Greenery
NMFsc	Tree, Grass, Pole, Building, People, Clear Sky, Sidewalk, Leafless Tree	Cloudy Sky, Stadium, Stand, Football Field, People, Band, Post, Track, Banner, Stands	Tree, Cloudy Sky, Building, Water, Dock, Powerboat, Hill, Ferryboat

FIGURE 6.13: Some sample results of image auto-annotation using the classic NMF (CNMF) and NMF with sparseness constraints (NMFsc).

Method	Number of Words	Precision	Recall	E_{NS}
Vector-Space	~ 4.8	0.476	0.465	0.450
	~ 7.42	0.402	0.581	0.554
	~ 9.70	0.350	0.641	0.602
LSI(K=40)	~ 4.8	0.490	0.480	0.466
	~ 7.42	0.414	0.588	0.561
	~ 9.70	0.356	0.648	0.609
CNMF	~ 4.8	0.478	0.465	0.450
	~ 7.42	0.403	0.585	0.557
	~ 9.70	0.352	0.651	0.612
NMFsc	~ 4.8	0.492	0.475	0.461
	~ 7.42	0.413	0.593	0.566
	~ 9.70	0.360	0.658	0.619

TABLE 6.2: Summary of results of image auto-annotation using several different semantic propagation-based methods.

segments when multiple segmentations are generated. We have obtained results that are comparable with those reported by Russell et al. (2006), who used more complicated statistical models.

The second part explored the use of NMF as an alternative sub-space approach to latent semantic indexing. Test images are mapped into the semantic space spanned by the basis vectors (columns of W) that are generated by NMF on the term-by-document matrix of the training set. For each test image, the closest training images in the space are used for semantic propagation. We have experimented with two versions of NMF, the classic NMF and NMF with sparseness constraints. In particular, NMF with sparseness constraints performed slightly better than SVD in terms of image auto-annotation through semantic propagation.

Therefore, we argue that NMF is a promising sub-space technique for discovering the latent structure of image data-sets, with the ability of encoding the latent topics that correspond to object classes in the basis vectors generated.

Chapter 7

The Image Based Feature Space Model

So far, a majority of the published image auto-annotation techniques only generate labels at the whole image level (globally) (Jeon et al., 2003; Feng et al., 2004; Tang et al., 2006), rather than at the object or region level (locally) (Duygulu et al., 2002; Yang et al., 2005). For example, as shown in Figure 7.1, captions that are assigned to image 7.1(a) and 7.1(b) are global, which carries only the information that certain objects exist in this image. In contrast, captions of images in Figure 7.2 are attached to specific regions, which provides the information about object location and extent. In other words, global image auto-annotation does not indicate which part of the image gives rise to which word, so it is not explicitly object recognition.

In most of the existing annotated image databases that are used for auto-annotation experiments, labels are associated with the whole images rather than individual regions, which makes region based image annotation a challenge. Detection of the location and extent of objects on such image data-sets, as well as its application to image auto-annotation is the problem to be explored in this chapter. The contents of this chapter have been published in (Tang and Lewis, 2007b) and (Tang and Lewis, 2007c).

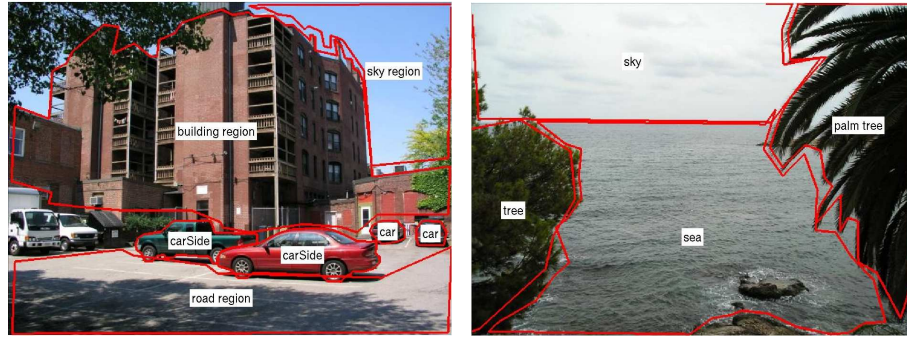
Firstly, an image based feature space (IBFS) model and mapping of image regions and labels are defined. The embedding of both image regions and textual labels into the same space makes object recognition more straightforward. The links between image regions and words are measured by their separation in the feature space, which is further used to annotate images, both locally and globally.

Secondly, the idea of using multiple segmentations for image auto-annotation is proposed. For each image, multiple layers of segmentation are generated and then incorporated into the IBFS model for automatic image annotation.



(a) Captions: Tree, Building, Grass, Sidewalk, Pole, People, Clear Sky
 (b) Captions: Stadium, Stand, People, Football Field, Band, Track, Banner, Flag, Lake, Tree, Partially Cloudy Sky

FIGURE 7.1: Examples of globally annotated images from the Washington data-set (University of Washington, 2004).



(a)

(b)

FIGURE 7.2: Examples of locally annotated images from the LabelMe data-set (Russell et al., 2006).

7.1 An Image Based Feature Space and Mapping

In this section, we try to discover the correspondence between image regions and textual labels. Specifically, given a collection of images that are only annotated globally, we want to find out which region of an image represents which word that annotates the image. An image based feature space and mappings of both image regions and textual labels into that space are defined. The links between image regions and words can be discovered from their separation in the feature space. It is then applied to image auto-annotation, both globally and locally. This section begins with an introduction of two techniques that are able to attach words to specific image regions. Then, the details of the algorithm will be described, followed by experimental results and some discussions.

7.1.1 Related Work

Duygulu et al. (2002) view the process of image auto-annotation as machine translation. They first used a segmentation algorithm to segment images into object-shaped regions,

followed by the construction of a visual vocabulary, which is represented by ‘blobs’. Then, a machine translation model is utilized to translate between ‘blobs’ comprising an image and words annotating that image. Thus, it is capable of annotating objects in images.

Yang et al. (2005) use Multiple-Instance Learning (MIL) (Maron and Lozano-Pérez, 1998) to learn the correspondence between image regions and keywords. “Multiple-instance learning is a variation on supervised learning, where the task is to learn a concept given positive and negative bags of instances.” (Maron and Lozano-Pérez, 1998). Labels are attached to bags (globally) instead of instances (locally). In their work, images are considered as bags and objects are instances. However, for each keyword only one representative region was learnt and used as the basis for determining new un-labeled regions. Intuitively, this approach is more or less restricted, because one concept could have very different sample regions, such as “flowers”, which can be of any colour and shape.

Although there are many other image auto-annotation techniques, to the best of our knowledge the above two are the only models that annotate regions. For example, Jeon et al. (2003) proposed a cross-media relevance model that learns the joint probabilities of a set of regions (blobs) and a set of words, instead of the one-against-one correspondence. In order to annotate a new image, they used all the regions within the image as a whole and chose the words with the highest joint probabilities. We argue that, at some level, models like this benefit from the fact that the data-set contains many globally similar images. As illustrated in (Tang and Lewis, 2006), a simple global feature descriptor based propagation method achieves even better results on the same data-set. Therefore, in this paper, we compare our approach with only the two region based annotation models mentioned above.

7.1.2 The Algorithm

7.1.2.1 Approach Overview

We propose an image based feature space and a mapping of image regions and words into the space for object recognition. Firstly, each image is segmented automatically into several regions. For each region, a feature descriptor is calculated. We then build a feature space, each dimension of which corresponds to an image from the database. Finally, we define the mapping of image regions and labels into the space. The correspondence between regions and words is learned based on their relative positions in the feature space. In terms of regional image annotation, a test region is annotated with the closest word in the space. Furthermore, regional labels that are most likely to be correct are used as the annotations for the whole image, for comparison with other approaches.

In the following, we first describe how image segments can be represented by visual terms which are based on salient regions. Secondly, we propose how to embed image regions and words into an image based feature space, in order to find the relationships between words and image regions. Then, its application to region-based image annotation will be described. Finally, a simple example is presented as an illustration of the algorithm.

7.1.2.2 Representing image regions by salient regions

There are very many different automatic image segmentation algorithms. In this work the Normalized Cuts framework (Shi and Malik, 2000) is used because it handles segmentation in a global way which has more chance than some approaches to segment out whole objects.

Once images are segmented, a descriptor is calculated for each image segment. The approach of Tang et al. (2006)'s work is followed to represent images by salient regions. Specifically, we first select salient regions by using the method proposed by Lowe (2004), in which scale-space peaks are detected in a multi-scale difference-of-Gaussian pyramid. Lowe's SIFT (Scale Invariant Feature Transform) descriptor is used as the feature descriptor for the salient regions. The SIFT descriptor is a three dimensional histogram of gradient location and orientation. The descriptor is constructed in such a way as to make it relatively invariant to small translations of the sampling regions, as might happen in the presence of imaging noise. Quantisation is applied to the feature vectors to map them from continuous space into discrete space. Specifically, the k -means clustering algorithm is adopted to cluster the whole set of SIFT descriptors. Each cluster then represents a visual word from the visual vocabulary. As a result, each image segment can be represented by a k -dimensional frequency vector or histogram, for the visual words contained within the segment.

7.1.2.3 Image-Based Feature Mapping

We define an image-based feature mapping \mathbf{m} , which maps each label and image segment into a feature space \mathbf{F} . The feature space \mathbf{F} is an N dimensional space, where N is the total number of images in the training set and where each dimension corresponds to a training set image.

Mapping of image segments: We denote images as I_i ($i = 1, 2, \dots, N$), and the j th segment in image I_i as I_{ij} . For the sake of convenience, we line up all the segments in the whole set of images together and re-index them as I^t ($t = 1, 2, \dots, n$, n being the total number of segments). The coordinates of a segment in \mathbf{F} are defined by the mapping \mathbf{m} :

$$\mathbf{m}(I^t) = [d(I^t, I_1), d(I^t, I_2), \dots, d(I^t, I_N)] \quad (7.1)$$

where $d(I^t, I_i)$ represents the coordinate of segment I^t on the i th dimension, which is either 1 or 0 according to the distance of I^t to image I_i . The distance of a segment to an image is defined as the distance to the closest segment within the image. The distance between two vectors/histograms V_1 and V_2 , which represent two segments, is measured by the normalised scalar product (cosine of angle), $\cos(V_1, V_2) = \frac{V_1 \bullet V_2}{|V_1||V_2|}$. A threshold t is set to decide if the coordinate is 1 or 0, as follows

$$d(I^t, I_i) = \begin{cases} 1 & \text{if } \max_{j=1, \dots, n_i} \cos(I^t, I_{ij}) > t \\ 0 & \text{otherwise} \end{cases} \quad (7.2)$$

The mapping of segments can be comprehended as a mapping in which, if the object that a segment contains also appears in a particular training image, the coordinate of the segment on the dimension represented by that image is 1, otherwise 0. Intuitively, segments relating to the same objects or concepts should be close to each other in the feature space.

Mapping of textual words: We can also map labels used to annotate the images into the space. Suppose the vocabulary of the data-set is W_l ($l = 1, 2, \dots, M$, M being the vocabulary size). The coordinate of a label on a particular dimension is decided by the image this dimension represents. If the image is annotated by that label, the coordinate is 1, otherwise it is 0. Therefore, the mapping of words is defined as:

$$\mathbf{m}(W_l) = [d(W_l, I_1), d(W_l, I_2), \dots, d(W_l, I_N)] \quad (7.3)$$

where

$$e(W_l, I_i) = \begin{cases} 1 & \text{if } I_i \text{ is annotated by } W_l \\ 0 & \text{otherwise} \end{cases} \quad (7.4)$$

Ideally, a label should be close to the image segments associated with the objects the label represents. The normalised scalar product is used to measure the distance between a segment and label, calculated as $\cos(\mathbf{m}(I^t), \mathbf{m}(W_l))$.

The mapping of segments and words in this way is similar to the work of Bi et al. (2005), in which a region-based feature mapping is used. However, they defined a feature space in which each dimension is an image segment, and then map each image into the space. In other words, the two mappings are essentially the inverse of each other. However, one of the advantages of our mapping is that it is also able to map image labels to the feature space. For the mapping of Bi et al. (2005), there is no way to identify the coordinate of a label on each dimension of the feature space because labels are only attached on an image basis, rather than a region basis. In addition, instead of using global features (colour, shape, texture), we use a histogram of visual words, which are quantised from salient regions within each image segment.

7.1.2.4 Application to Region-Based Image Annotation

Similarly, the segments of test images can also be mapped into the training image based feature space as described above. Region based image annotation becomes relatively straightforward once the mapping is done. The probability of a segment being correctly annotated by a particular label, is approximated by their cosine distance in the space. For each test segment, the word with the highest probability is chosen. In terms of global image annotation, the set of words that are generated from all the segments within the image are lined up and those with the highest probabilities are chosen.

7.1.2.5 A Simple Example

In this section a simple example is presented to illustrate the major steps of the method. Consider two annotated images; I_1 is labelled as “RED, GREEN” and half of the image is red and the other half is green; I_2 is labelled as “GREEN, BLUE” and half is green and the other half is blue, as shown in Figure 7.3. Assume the segmentation algorithm manages to separate the two colours in each image and segments them into halves. We will have four segments in all, denoted as I^1, I^2, I^3 and I^4 . Using the RGB values as the feature descriptors, the segments can be represented as $I^1 = (255, 0, 0), I^2 = (0, 255, 0), I^3 = (0, 255, 0), I^4 = (0, 0, 255)$. Then we need to map the segments into the feature space, which is a two dimensional space in this case as there are two images. By applying Equation 7.1, the coordinates of the segments are as follows:

$$\begin{aligned} I^1 : & [1, 0]; \\ I^2 : & [1, 1]; \\ I^3 : & [1, 1]; \\ I^4 : & [0, 1]; \end{aligned} \tag{7.5}$$

The step of applying the quantisation equation 7.2 is omitted here, since the coordinates calculated are already 0 and 1s. Alternatively, a threshold $t \in (0, 1)$ can be set for the same results. In addition, the labels can also be mapped into the feature space to give:

$$\begin{aligned} RED : & [1, 0]; \\ GREEN : & [1, 1]; \\ BLUE : & [0, 1]; \end{aligned} \tag{7.6}$$

It can now be seen that in the feature space, the closest labels for the segments are:

$$\begin{aligned} I^1 : & RED; \\ I^2 : & GREEN; \\ I^3 : & GREEN; \\ I^4 : & BLUE; \end{aligned} \tag{7.7}$$

Refer to Figure 7.3 for a diagram of this example.

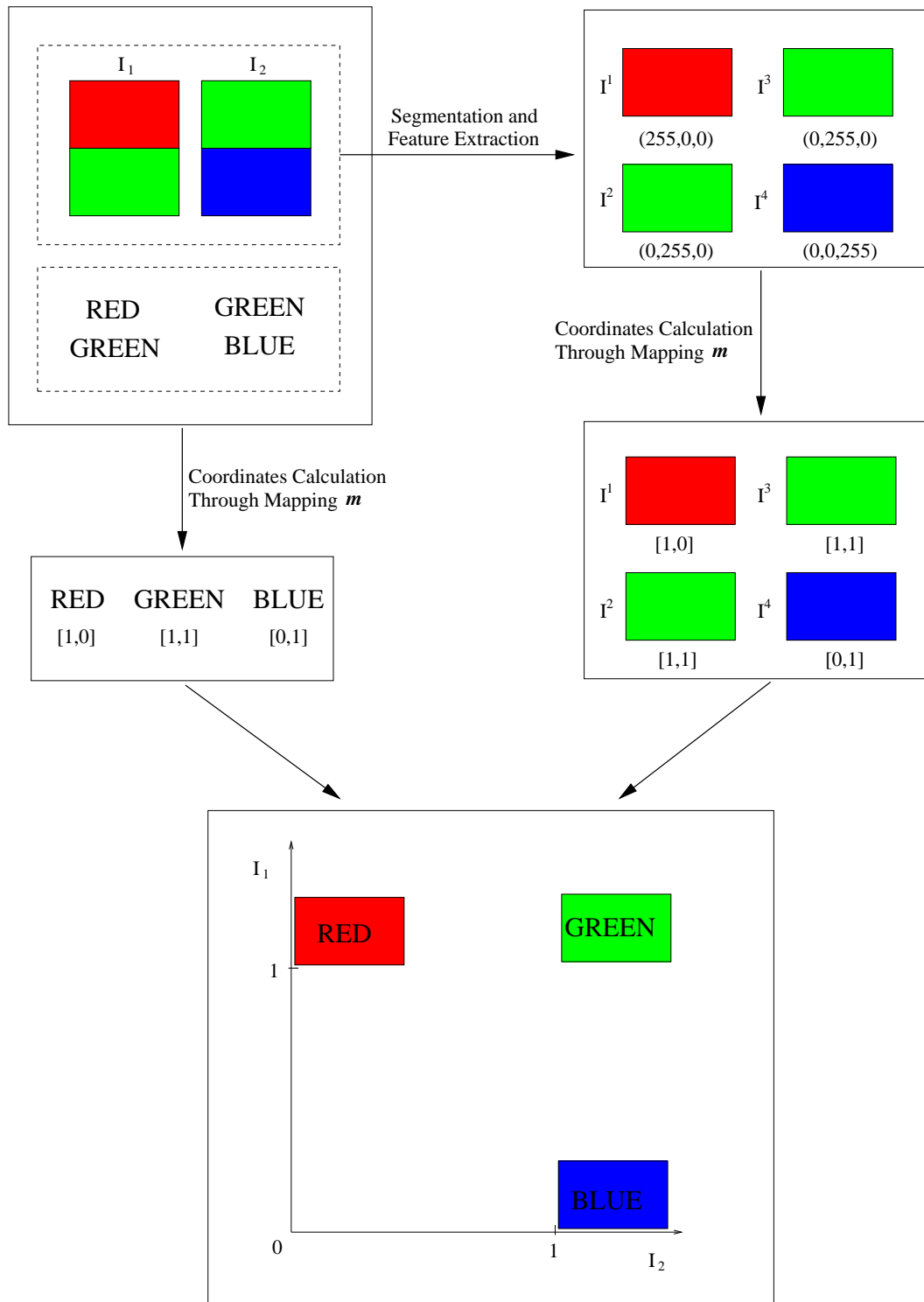


FIGURE 7.3: Diagram of a simple example of using the image based feature space for relating image regions and words

7.1.3 Experimental Results and Discussion

7.1.3.1 Correspondence of segments and words

The experiments in this section aim to demonstrate the effectiveness of this approach in finding the correspondence between image segments and words. The method has been applied to the Washington image set¹ which contains 697 semantically annotated images. After the original keyword labels were processed by correcting mistakes and merging plurals into singular forms (Hare and Lewis, 2005c), the vocabulary consisted of 170 keywords. The whole set of SIFT descriptors are quantized into 3000 visual words. The number of segments is set to 5 per image when using Normalized Cuts (Shi and Malik, 2000). This results in 3241 segments after removing those having no salient regions within them. We build a training image based feature space, into which all the image segments and words are mapped afterwards. For each keyword, we find in the feature space the 25 closest segments. The number of correct segments for each keyword is counted manually and those for the 25 keywords (Figure 7.4) with the highest occurrences in the data-set are reported in Table 7.1. Because the original labels are only attached to the whole image, the decision of whether a segment is correct or not is made by human judgement. We consider a segment as correct if the corresponding object occupies more than 50 percent of the area of the segment, otherwise not. Figure 7.8 shows some good examples, and Figure 7.9 shows some bad ones.

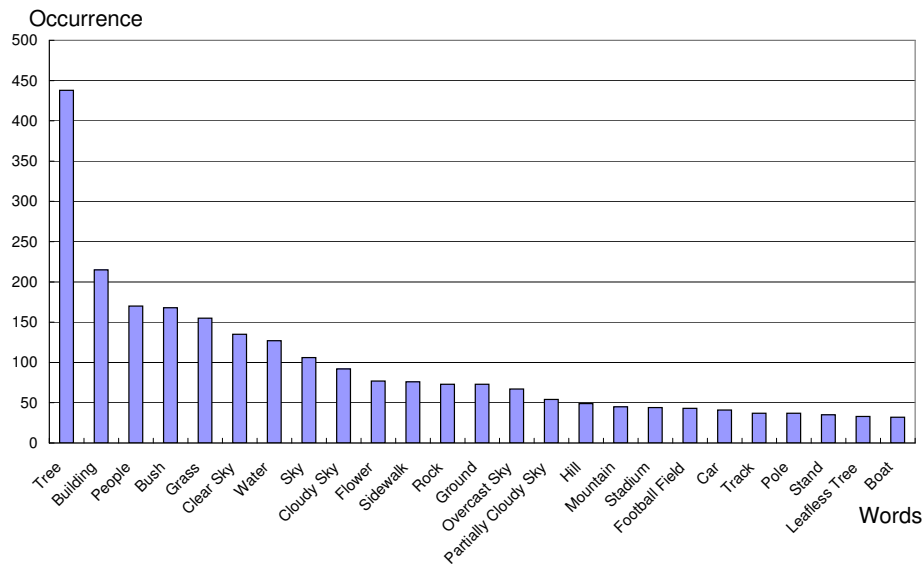


FIGURE 7.4: Top 25 Words that appear most frequently in the Washington set

As shown in Table 7.1, the results for some keywords (e.g. Water, Stadium, Building, Bush, Tree, etc.) are reasonably good, however, for the others they are less so. There are several possible explanations.

¹Available at: <http://www.cs.washington.edu/research/imagedatabase/groundtruth/>

1. First of all, some objects are too small in the image to be segmented out reliably by a 5 region N-cut. For example, “Sidewalk”, “Car” and “Boat” usually occupy small areas of the image in the Washington set and rarely achieve good segmentations, as shown in Figure 7.5. Therefore, the algorithm returns the objects that have a high co-occurrence with these words. For “Sidewalk”, image segments with “Trees” are found; For “Car” and “Boat”, segments with “Building” and “Water” are found respectively, as shown in Figure 7.9.
2. Secondly, some words occur together almost every time they occur and rarely occur separately. This is analogous to an extreme example where a child who has never learnt what a knife and fork look like, is given many images in which both knife and fork appear together, even if he/she is told that all the images contains a knife and fork, there is no way for the child to learn which is which. In the Washington set, “Football Field”, “Track” and “Stand” co-occur almost totally. As shown in Figure 7.6, for each cell, the number on the dashed line indicates the number of times two words appear together (in the same image), and the other two numbers indicate their occurrence alone without the other. For example, “Track” and “Football Field” occur 36 times together, but only 1 and 7 times respectively on their own. Because of high co-occurrence, the algorithm failed to distinguish them from each other. Almost the same results are returned for them, mostly “Football Field” as shown in Figure 7.8(c), which is probably because the feature descriptors for “Football Field” are more stable.
3. Lastly, insufficient feature descriptors. Since the SIFT descriptor is using only grey level information, objects that are mainly distinguished by colour will be hard to identify. For example, in this work, the segments returned for “Flower” contain a lot of “Tree” labels (Figure 7.9(d)), probably because in the data-set, the SIFT feature descriptors for both “Flower” and “Tree” are similar and also often co-occur as well. For example, as shown in Figure 7.7, it is easy to tell “Flower” from the top colour images, but hard to distinguish from the corresponding gray images at the bottom.

7.1.3.2 Results on Region Based Image Annotation

In this section, we compare the effectiveness of the image based feature space (IBFS) approach in region based image annotation with other approaches. For fair comparison, we used the same data-set² as that used in the experiments of Duygulu et al. (2002); Yang et al. (2005) on region based image annotation. The dataset contains 5000 images from 50 Corel Stock Photo CDs, and has been divided into a training set of 4500 images and a test of 500 images. Each image had been annotated manually with 1-5 keywords.

²Available at: http://kobus.ca/research/data/eccv_2002/index.html



(a) Segmentation samples of images containing “Sidewalk”



(b) Segmentation samples of images containing “Boat”



(c) Segmentation samples of images containing “Car”

FIGURE 7.5: Segmentation samples of images from the Washington data-set (University of Washington, 2004).

	Track	Stand	Football Field
Track	5	30	36
Stand	7	1	9
Football Field	7	34	1

FIGURE 7.6: The number of times words “Track”, “Stand” and “Football Field” occur together and separately.

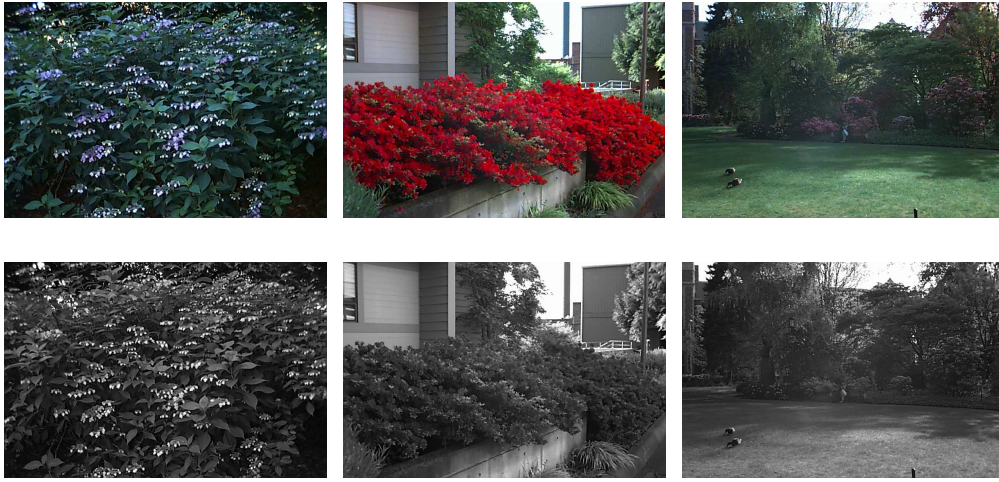


FIGURE 7.7: Samples of colour images containing “Flower” and the counterpart gray ones from the Washington data-set (University of Washington, 2004). The top ones are RGB colour images, and the bottom ones are the corresponding gray images.

Images are segmented by Normalised Cut (Shi and Malik, 2000), which generated 5-10 regions per image. Each region is then represented by a 30 dimensional feature vector, including region average colour, size, location, average orientation energy and so on, as described in (Duygulu et al., 2002). Feature vectors are normalised to Z-Scores for distance measure in the image based space. Specifically, suppose the whole set of training feature vectors are V_1, V_2, \dots, V_n , n being the total number of training image segments, and $V_i = \{V_{i1}, V_{i2}, \dots, V_{i30}\}$, we calculate the Z-Score for the j th dimension of the i th vector as follows

$$Z_{ij} = \frac{V_{ij} - \text{mean}(V_{1j}, V_{2j}, \dots, V_{nj})}{\text{standard deviation}(V_{1j}, V_{2j}, \dots, V_{nj})} \quad (7.8)$$

Feature vectors of the test set are also normalised, using the mean and standard deviation of the training vectors.

In order to find the optimal value for threshold t in Equation (7.2), 500 random images are taken out of the training set for evaluation, by training on the remaining 4000 images. Based on the average per-word precision and recall that are calculated on the evaluation

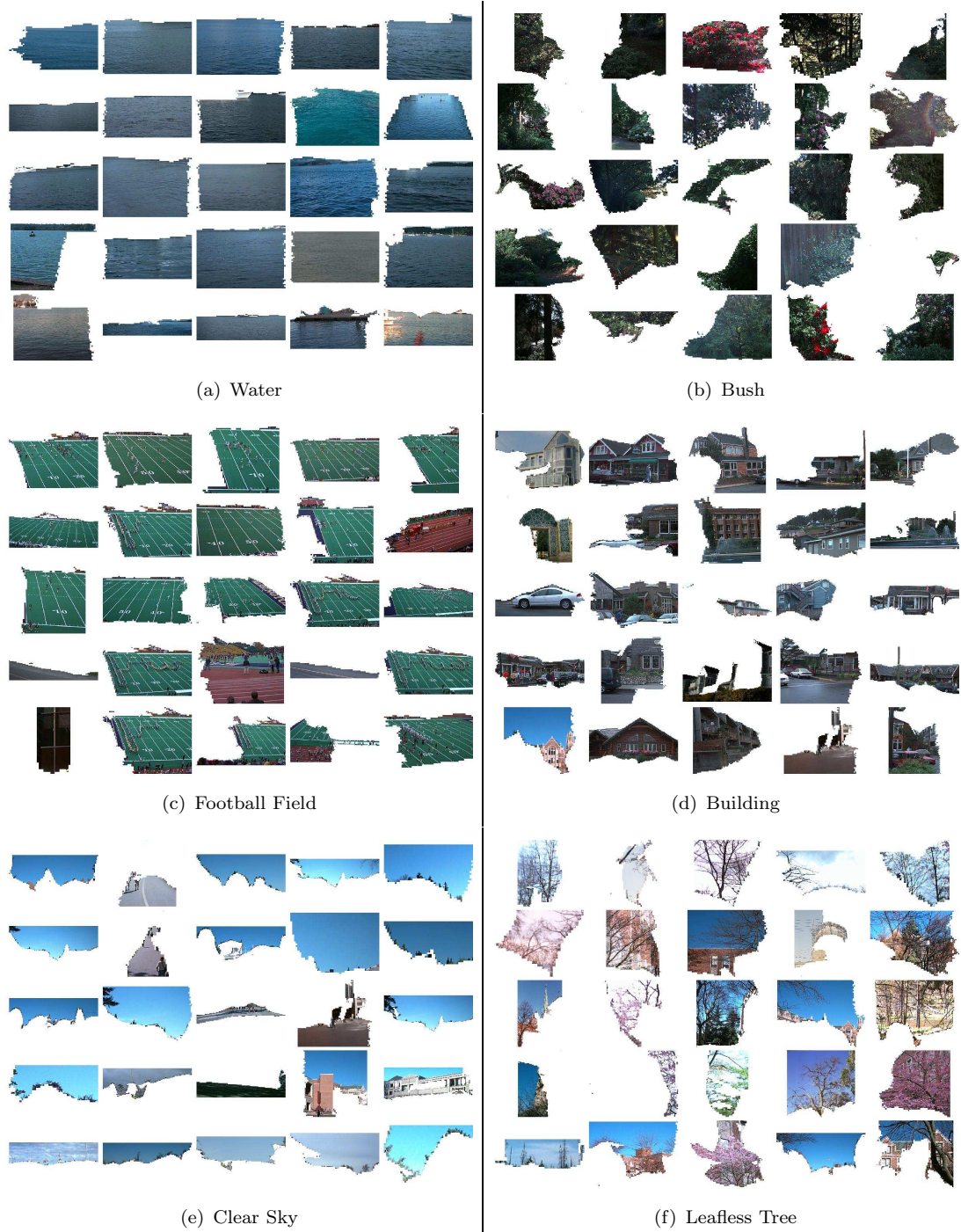


FIGURE 7.8: Some good results of representative regions found by the image based feature space approach for the corresponding words.

Keywords	Our Method	Random
Tree	21	3
Building	22	0
People	21	1
Bush	25	3
Grass	6	1
Clear Sky	19	0
Water	25	2
Sky	19	3
Cloudy Sky	21	3
Flower	8	2
Sidewalk	2	0
Rock	6	2
Ground	6	3
Overcast Sky	0	3
Partially Cloudy Sky	20	4
Hill	0	2
Mountain	5	1
Stadium	22	0
Football Field	20	1
Car	7	1
Track	2	0
Pole	1	0
Stand	0	1
Leafless Tree	24	1
Boat	0	0

TABLE 7.1: The number of correct segments out of the top 25 for our method and random choice.

set, the best performance is found at $t = 0.88$ (with step of 0.01). Thereafter, the 4500 training images are used to build the feature space, into which the test image segments are mapped. For each test image segment, the word with the highest probability is chosen as the region label. Figure 7.10 shows some good examples, while Figure 7.11 shows some bad examples. Note that for some regions, the coordinates on all dimensions of the space are zeros (i.e. $\mathbf{m}(I^t) = [0, 0, \dots, 0]$) after applying quantisation equation 7.2. It implies that the IBFS technique regards such regions as not similar to any of the training image regions available. In such cases, we assign the word “null” to these regions, indicating that the approach can not decide which word to attach.

In order to compare with other image annotation techniques, we predict image based labels by choosing words from the region labels. For each test image, the top 5 region labels with the highest values of probability are chosen. We compare our approach with two region based image annotation approaches mentioned in section 7.1.1, namely the Machine Translation Model (Duygulu et al., 2002) and the Point-wise Diverse Density Model (Yang et al., 2005). For a direct comparison, average per-word precision and recall are calculated, as described in section 4.3, for the whole set of words and the best

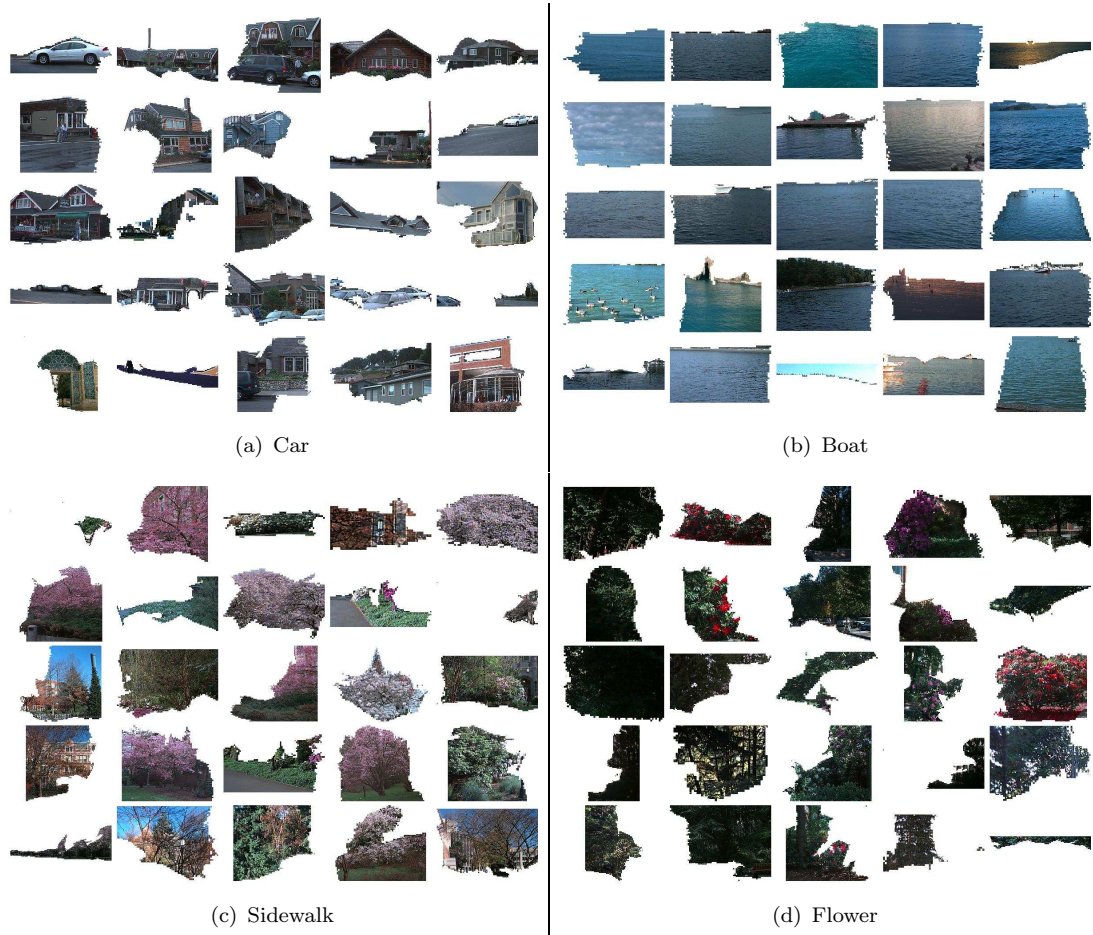


FIGURE 7.9: Some bad results of representative regions found by the image based feature space approach for the corresponding words.

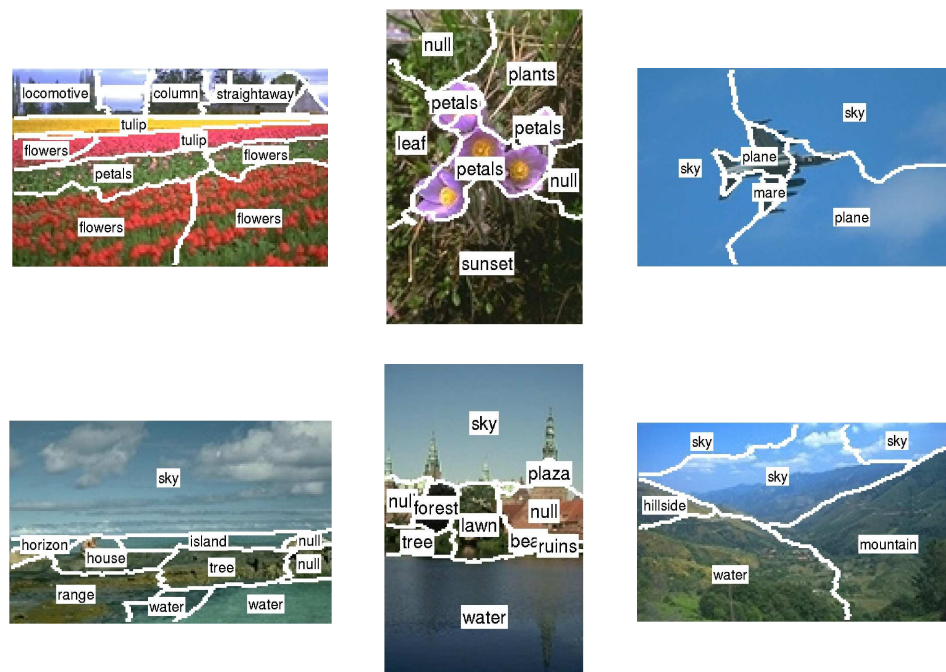


FIGURE 7.10: Some good examples of image region annotation through the image based feature space approach.

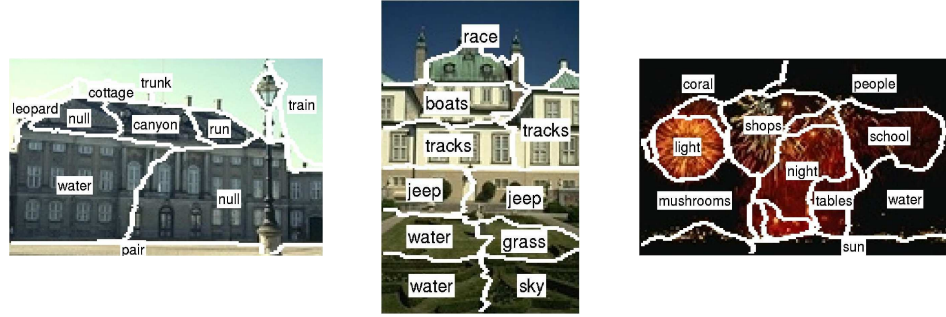


FIGURE 7.11: Some bad examples of image region annotation through the image based feature space approach.

Approaches	All Words		Best 49 Words	
	Avg. pr.	Avg. re.	Avg. pr.	Avg. re.
MT	0.04	0.06	0.20	0.34
PWDD	0.07	0.09	0.31	0.46
IBFS	0.10	0.11	0.39	0.51

TABLE 7.2: Performance comparison of Machine Translation Model (MT), Point-wise Diverse Density Model (PWDD) and Image Based Feature Space Model (IBFS).

49 words (for comparison with the work by Duygulu et al. (2002); Yang et al. (2005)). As shown in Table 7.2, the image based feature space (IBFS) method achieves better results on the same data-set.

7.1.4 Summary

A novel training image based feature space has been proposed together with a procedure for mapping in both image segments and textual labels for region based image annotation. Some segments associated with the same objects should be clustered together, and also close to the label that represents the object in question. As a result, the relationships between image regions and words can be discovered by comparing their distances in the feature space. Annotating new image segments and images is also straightforward. Regional labels for test images are predicted by choosing the words that are closest in the space. Furthermore, region labels that are most likely to be correct are adopted as the global labels for the whole image.

7.2 Multiple Segmentations for Image Auto-Annotation

Automatic image annotation techniques that try to identify the objects in images usually need the images to be segmented first, especially when specifically annotating image regions. The purpose of segmentation is to separate different objects in images from each other, so that objects can be processed as integral individuals. Considering the massive

work load of manual segmentation, most researchers rely on automatic segmentation techniques (Deng et al., 1999; Shi and Malik, 2000). Therefore, annotation performance is highly influenced by the effectiveness of segmentation. Unfortunately, automatic segmentation is a difficult problem, and most of the current segmentation techniques do not guarantee good results. A multiple segmentations algorithm is proposed by Russell et al. (2006) to discover objects and their extent in images. In this section, we explore the novel use of multiple segmentations in the context of image auto-annotation. It is incorporated into the region based image annotation technique proposed in the previous section.

7.2.1 The Algorithm

7.2.1.1 Approach Overview

Russell et al. (2006) propose to use multiple segmentations to discover objects and their extent in images. They vary the parameters of a segmentation algorithm in order to generate multiple segmentations for each image. They do not expect any of the segmentations to be totally correct, but “the hope is that some segments in some of the segmentations will be correct”. Then, topic discovery models from statistical text analysis are introduced to analyze the segments, in order to find the good ones. Their approach managed to find the correct image segments more successfully than using a single segmentation.

Inspired by Russell et al. (2006)’s work, we propose to incorporate the idea of multiple segmentations into automatic image annotation. Within a large image data-set, the good segments of the same object will share similar visual features, but the bad ones will have random features of their own. As Russell et al. (2006) said “all good segments are alike, each bad segment is bad in its own way”. We hope that by using multiple segmentations, more good segments can be generated (although from different segmentations), and then captured by auto-annotation models in one way or another.

We chose to embed multiple segmentations into the image based feature space model. There are a few reasons to make this choice of model. Firstly, it is a region based annotation method, which is different from those that only annotate the whole images. Secondly, it is easy to implement, and achieves relatively good results. Lastly, transfer from single segmentation to multiple segmentations is more straightforward in this model - what needs to be done is just mapping more segments (i.e. segments from different segmentation levels) into the space, without changing the structure or dimensionality.

7.2.1.2 Generating Multiple Segmentations

There are very many different automatic image segmentation algorithms. In this work the Normalized Cuts framework (Shi and Malik, 2000) is used, following the choice of Russell et al. (2006), because it handles segmentation in a global way which has more chance than some approaches to segment out whole objects. In order to produce multiple segmentations, we varied one parameter of the algorithm, namely the number of segments K . Figure 7.12 shows some examples of segmented images at different levels of segmentation ($K = 4, 6$ and 8). Evidently, some objects (polar bear, pyramid) get better segmentation at a low level (i.e. a small number of segments), while others (zebra, flower) do so at a high level. However, almost all the objects get reasonably good segmentation at one of the levels, although not at the same one.



FIGURE 7.12: Examples of segmented images at different levels of segmentation

7.2.1.3 Incorporating Image Based Feature Space and Mapping with Multiple Segmentation

In section 7.1, we have shown the effectiveness of a training image based feature mapping in finding representative regions for labels, as well as in region based image auto-annotation. However, in both cases a single level of segmentation is used. In this work, we incorporate the image-based feature mapping approach with the idea of multiple segmentations, in order to take into consideration the fact that different objects have their best segmentation at different levels. The details of mapping for textual words is not given here, because it is the same as that for single segmentation which can be found in section 7.1.2.3. However, mapping for segments from multiple levels is different as detailed in the following.

We denote training images as I_i ($i = 1, 2, \dots, N$, N being the total number of images), and the j th segment in image I_i at the k th segmentation level as I_{ikj} . For the sake of convenience, we line up all the segments from all the segmentation levels of the whole set of training images together and re-index them as I^t ($t = 1, 2, \dots, n$, n being the total number of segments).

The coordinates of a segment I^t in \mathbf{F} are defined as:

$$\mathbf{m}(I^t) = [d(I^t, I_1), d(I^t, I_2), \dots, d(I^t, I_N)] \quad (7.9)$$

where $d(I^t, I_i)$ represents the coordinate of segment I^t on the i th dimension, which is either 1 or 0 according to the distance of I^t to image I_i . The distance of a segment to an image is defined as the distance to the closest segment within all the segmentation levels of the image. By comparing segments from all levels, we hope that good segments from different segmentations can be matched, which is less likely when single segmentation is used. The distance between two vectors/histograms V_1 and V_2 , which represent the feature descriptors of two segments, is measured by the normalised scalar product (cosine of angle), $\cos(V_1, V_2) = \frac{V_1 \bullet V_2}{|V_1||V_2|}$. A threshold t is set to decide if two segments are close enough or not, which then generates either 1 or 0 as the coordinate on one dimension of the space. Mathematically it is defined as follows

$$d(I^t, I_i) = \begin{cases} 1 & \text{if } \max_{k=1, \dots, m} (\max_{j=1, \dots, n_{ik}} (\cos(I^t, I_{ikj}))) > t \\ 0 & \text{otherwise} \end{cases} \quad (7.10)$$

where m is the number of segmentation levels, n_{ik} is the number of segments of image I_i at level k . The mapping of segments can be comprehended as a mapping in which if the object that a segment contains also appears in a particular training image, the coordinate of the segment on the dimension represented by that image is 1, otherwise 0.

To annotate test images, all the test segments from all levels are mapped into the training image based feature space. The test set is denoted as $T_{i'}$ ($i' = 1, 2, \dots, N'$, N' being the total number of test images), and the j' th segment from the k' th level of image $T_{i'}$ is denoted as $T_{i'k'j'}$. All the test segments are lined up and denoted as $T^{t'}$. By applying the mapping \mathbf{m} to a test segment $T^{t'}$, we can calculate its coordinates in the training image based space as follows

$$\mathbf{m}(T^{t'}) = [d(T^{t'}, I_1), d(T^{t'}, I_2), \dots, d(T^{t'}, I_N)] \quad (7.11)$$

Region based image annotation becomes relatively straightforward once the mapping is done. The probability of a segment being correctly annotated by a particular label, is approximated by their distance in the space. Furthermore, the probability of a test image being correctly annotated by a label, $P(W_l, T_{i'})$, is estimated by the highest probability

Feature of Region	Dimension
Area	1
Position	2
Boundary/Area	1
Convexity	1
Moment of Inertia	1
Average RGB	3
RGB Stdev	3
Average L*a*b	3
L*a*b Stdev	3
Mean Oriented Energy	12

TABLE 7.3: Region features

of this label being correct with any of the segments in that image, as follows

$$P(W_l, T_{i'}) = \max_{k'=1, \dots, m'} (\max_{j'=1, \dots, n_{i'k'}} (\cos(\mathbf{m}(W_l), \mathbf{m}(T_{i'k'j'})))) \quad (7.12)$$

where m' is the number segmentation levels, while $n_{i'k'}$ is the number of segments of test image $T_{i'}$ at level k' . Finally, words with highest value of $P(W_l, T_{i'})$ are chosen as the predicted captions of the image.

7.2.2 Experiment and Results

Section 7.1 has already demonstrated that the training image based feature space technique outperforms two other state of the art region based image auto-annotation techniques (Duygulu et al., 2002; Yang et al., 2005). Other image auto-annotation techniques were not considered because to the best of our knowledge, none of them is able to annotate image regions.

In this work, we compare the effectiveness of using multiple segmentations for image auto-annotation with that of single segmentation. The same image collection (Duygulu et al., 2002; Yang et al., 2005; Tang and Lewis, 2007b), as used in section 7.1 is adopted for the experiment. We used Normalised Cut (Shi and Malik, 2000) and varied the parameter of segment number to generate multiple segmentations for each image. In this work, three levels of segmentation are set, 4, 6 and 8. Therefore, the approaches we are comparing are one multiple segmentation approach (denoted as Multi-Seg), which includes three levels 4, 6 and 8, and three single segmentation ones (denoted as 4-Seg, 6-Seg and 8-Seg).

We follow Duygulu et al. (2002)'s representation of regions, which is a 30 dimensional feature vector, including region average colour, size, location, average orientation energy and so on, as detailed in Table 7.3. Feature vectors are normalised to Z-Scores using equation 7.8 for distance measure in the image based space. Note that for multiple segmentations, mean and standard deviation are calculated over the feature vectors

Approaches	Avg. pr.	Avg. re.
4-Seg	0.103	0.129
6-Seg	0.106	0.128
8-Seg	0.100	0.127
Multi-Seg	0.107	0.139

TABLE 7.4: Performance comparison of using multiple segmentations for image auto-annotation with single segmentation

from all segmentation levels, while for single segmentation, they are calculated within each segmentation level. Feature vectors of the test set are also normalised, using the mean and standard deviation of the training vectors.

In order to find the optimal value for threshold t in Equation (7.2) for each approach, 500 random images are taken out of the training set for evaluation, by training on the remaining 4000 images. Thresholds with the best performances are chosen for the actual auto-annotation experiment. For each test image, the top 5 labels with the highest values of probability are chosen, according to Equation (7.12).

The *Mean Per-word Precision and Recall*, as used by previous researchers (Duygulu et al., 2002; Jeon et al., 2003; Feng et al., 2004; Carneiro and Vasconcelos, 2005; Yang et al., 2005), are adopted for evaluation. As shown in Table 7.4, Multi-Seg achieves the best results. In addition, for each approach, we varied the threshold t from 0.99 to 0.80 with step of 0.01 to make further comparisons. *Keyword Number with Recall > 0* and *total correct number of words* are evaluated. A keyword has recall > 0 if it is predicted correctly once or more, otherwise not. As shown in Figure 7.13, Multi-Seg managed to predict the most number of keyword with recall > 0 ($t = 0.97$) among all the approaches. Moreover, as shown in Figure 7.14, Multi-Seg managed to predict more correct words than all the single segmentation approaches. In Figure 7.15, we present some annotation results of the multiple segmentations based approach.

7.2.3 Summary

We have proposed a way of coupling multiple segmentations with image auto-annotation. The parameter of segmentation algorithm is varied to generate several levels of segmentation. On the other hand, a region based image annotation approach, namely the image based feature space, is utilized to incorporate with multiple segmentations. We have shown that annotation performance can be improved on a 5000 image collection when multiple segmentations are used.

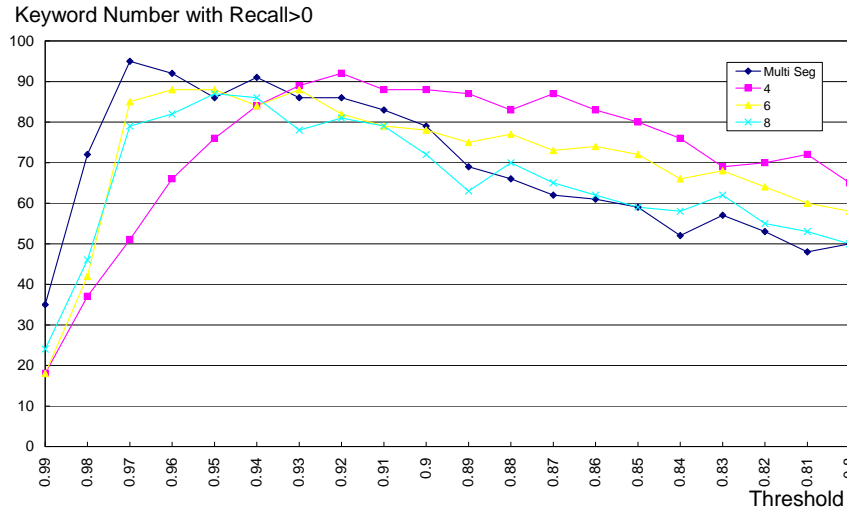


FIGURE 7.13: The number of keywords with recall>0 for each approach at different values of threshold

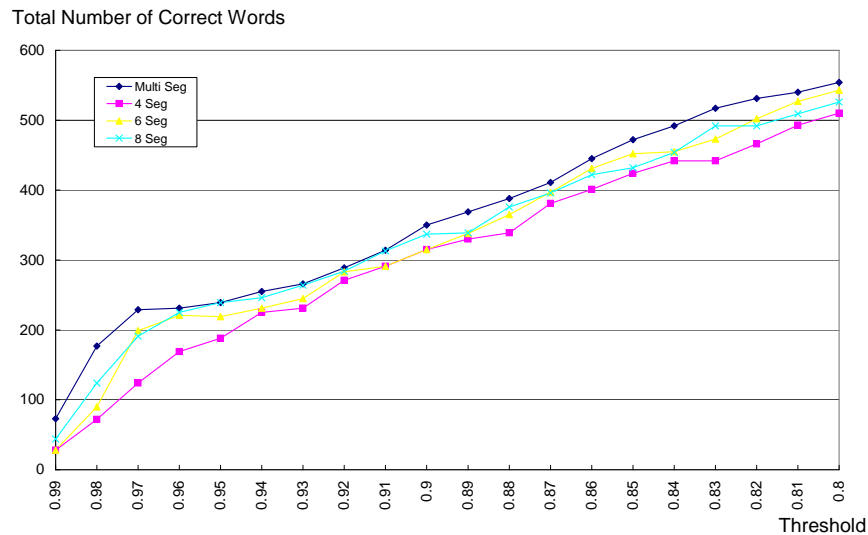


FIGURE 7.14: The total number of correctly predicted words for each approach at different values of threshold

7.3 Summary

In this chapter, we have explored two main techniques regarding object detection of images. The first method builds an image based feature space and then plots both image segments and labels into the space. The correspondences between image segments and labels are approximated by their separation in the space. Local and global image annotation can be realised in a straightforward way in this framework. This technique is later combined with a multiple segmentations approach. It has been demonstrated that the use of multiple segmentations achieves better results than that of single segmentation.

				
Original	clouds, sun, tree, water	plane, jet, sky	buildings, clothes, shops, street	flowers, garden, monks, people
Multi Seg Annotation	sun, tree, water, jeep, sky	plane, jet, sky, water, snow	people, street, cars, shops, buildings	flowers, people, petals, garden, nest

FIGURE 7.15: Some annotation examples by multiple segmentation based approach

Chapter 8

Conclusions and Future Work

This thesis has introduced a number of approaches and techniques for automatic image annotation, which is a subject that is receiving rapidly increasing attentions in recent years. One of the most important aims of automatic image annotation is to bridge the *semantic gap*, which is considered as a vital problem existing in the traditional content-based image retrieval systems. In this final chapter, we will draw together and summarise the main conclusions of the research undertaken by the author, and give some pointers to future research following the work presented in the earlier chapters.

8.1 Summary and Conclusions

Although automatic image annotation is a relatively new field of research, a lot of efforts have already been devoted by researchers to advance the technology. Chapter 3 to 7 have described a number of potential approaches to automatic image annotation, encompassing a variety of techniques regarding computer vision, machine learning and information retrieval.

Computer vision techniques that build up low-level image descriptors for describing and comparing image content are the foundation of research in automatic image annotation. The choice of an appropriate description mechanism has a large influence on the performance of an auto-annotation system. In chapter 3, we have introduced a number of such techniques. Image description is analysed as a 3-step process; region choosing, feature extraction and feature quantisation. Region choosing can be achieved by different means, from simple fixed partitioning to more advanced automatic segmentation and saliency detection approaches. Compared with fixed partitioning, automatic segmentation is expected to produce partitioning that is more consistent with the interpretation of images by humans, i.e. object-shaped regions. However, as automatic segmentation is “not just a bottom-up image processing problem, but also a top-down problem

that requires knowledge of the true object” Hare and Lewis (2005c), the performance of current auto-segmentation algorithms are very limited. Saliency detection methods are gaining more and more popularity in recent years, because they have been proved by many researchers to be capable of finding image regions that produce robust image descriptions.

The follow-up step is feature extraction, which actually calculate values from the images or image regions. Colour, shape and texture are three traditional features, each of which can be calculated in various ways. The SIFT local descriptor is robust and popular for describing salient regions. Feature quantisation is the final and optional step. It is usually adopted when the number of descriptors extracted from an image is too large to be handled by auto-annotation systems. Clustering algorithms are widely used for grouping the similar descriptors into classes, so that image can be represented by the memberships of the descriptors. The last section of chapter 3 described two practical examples of image description which are used in this work. The first one is the “blobs” representation using auto-segmentation algorithms and quantised traditional features, while the second is the “visual terms” representation using saliency and quantised salient regions descriptors.

The purpose of Chapter 4 is to examine some quality issues of the Coral data-set for evaluation on auto-annotation. By using the vocabulary of the Coral training set, we constructed a new data-set from Yahoo Image Search, named the Yahoo training set. Three very different auto-annotation methods were then applied to both data-sets. The results showed that the performance of a simple propagation based method (CSD-Prop) was fairly good on the Coral training set, but decreased dramatically on the Yahoo training set. In contrast, a support vector machine based method (CSD-SVD) performed more consistently on both sets. A closer look at the results regarding word combinations found that, CSD-SVD managed to predict combinations of words that do not exist in the training set but CSD-Prop can not. It is argued that the Coral set is relatively easy to annotate and contains too many very similar images, which may lead to biased results for some approaches, such as propagation-based methods.

Chapter 5 proposed a novel approach that incorporates the salient region based “visual terms” image representation into a statistical model, namely the Cross-Media Relevance Model (CMRM). A great number of statistical approaches to image auto-annotation use feature descriptors calculated from image segments, in the form of “blobs”. Our approach is different in that “visual terms” calculated from salient regions are used. One of the advantages of this approach is that the fallible step of segmentation can be avoided. The experimental results also showed that it achieved better performance than that of using “blobs”.

Chapter 6 was built upon the technique called non-negative matrix factorisation (NMF), which approximates a matrix with the product of two matrices containing all non-negative values. The first section is concerned with object class or topic discovery among a set of unannotated images. The “visual terms” image representation was used to build a term-by-document matrix. Thanks to NMF’s parts-based representation characteristic, the basis vectors obtained from the matrix decomposition process correspond to object classes in the data-set as shown in our experiments. NMF managed to capture object classes that occur frequently in the data-set by just looking at their visual appearances. We evaluated how well NMF can find the true extent of objects, and obtained results that are competitive with those reported by Russell et al. (2006), who used more complicated statistical models.

In the second section of chapter 6, we proposed and demonstrated the use of NMF as an alternative approach to latent semantic indexing. NMF is compared with singular value decomposition (SVD) in the experiments on image auto-annotation through semantic propagation. Again, NMF is applied to the term-by-document matrix built from the “visual terms ” representation. Each of the basis vectors (columns of W , see Equation 6.1.1.2) is considered as an axis in the sub-space. The similarities of Images are then measured by their relative positions in this space. In our experiments on the same data-set, it has been shown that NMF managed to generate auto-annotation results that are slightly better than SVD. Therefore, we argue that NMF is a sub-space technique that not only performs as well as SVD, but also supplies the advantage of parts-based representation.

Finally, in chapter 7, we took a step forward to annotate image regions, rather than the whole images. we proposed a novel model, the image based feature space model, for linking keywords to specific image regions. Training images are used to build the space, each image corresponding to an axis. Both image segments and labels are mapped into this space, in which their relationships can be measured. Visual terms are extracted from the images for building descriptors for image regions, the boundaries of which are decided by an auto-segmentation algorithm. By using the cosine distance between a keyword and image segment within the space as an approximation of the probability of the keyword given that segment, this model can be applied for image annotation at both local and global levels.

Later in this chapter, a multiple segmentation approach to automatic image annotation is presented. Under the assumption that multiple levels of segmentation have more chances to generate accurate segments, we argued that annotation results can also be improved by using multiple segmentations. We varied the parameters of the auto-segmentation algorithm to obtain multiple segmentations, and combined it with the image based feature space model developed earlier in the chapter. Incorporating multiple segmentations into this model is as straightforward as just mapping more image segments into the space.

Experimental results confirmed that annotation performance is enhanced. It also implies that the accuracy of segmentation is very important to an auto-annotation system. Improving the process of segmentation can lead to improvement in system performance.

8.1.1 Novel work of the thesis

This thesis has made a number of contributions to the community of automatic image annotation as well as object detection. A list of the contributions have been outlined in the introduction. In the following, we reaffirm the novelties associated with these contributions.

- Examination of the Coral data-set for evaluation through comparisons with another data-set constructed from an online image search engine.
- Incorporation of salient region descriptors into a statistical model for better auto-annotation performance.
- Application of NMF to general purpose images for object class detection.
- Demonstration of NMF as a promising alternative sub-space technique for image auto-annotation.
- Development of a model for linking image regions and labels through their positions in the same feature space.
- Development of a multiple segmentation approach to automatic image annotation.

8.2 Future Work

The thesis has presented a number of new ideas and techniques, which is just a snapshot of the on-going research undertaken by the author. In this section some directions for future research will be discussed on a chapter by chapter basis.

8.2.1 Image Description

Although image description is not the focus of this thesis, advancing the technology in this area will be fruitful for the performance of an auto-annotation system. There are a number of possible ways to improve the “blobs” image representation. Since it uses an automatic segmentation algorithm for region choosing, effectiveness of the segmentation method plays a very important role in the quality of the final representation. Using a more strong segmentation algorithm is likely to bring improvement. On the

other hand, because very simple image features have been extracted for describing each image segments, there is a lot of scope for improvement. The author is particularly interested in replacing them with the standardized MPEG-7 descriptors, as they are selected from many similar kinds of descriptors through a strict evaluation procedure, and are recommended as of high performance (Martinez, 2004). Similar possibilities for improvement exist in the “visual terms” representation. So far, a great number of salient region detection algorithms have been developed. An effective salient region detector will obviously contribute to the whole system. It has also been argued that for optimal performance of salient regions, the outputs of multiple salient region detectors should be combined (Mikolajczyk and Schmid, 2005). It would be interesting to investigate if the combination of different detectors will lead to a more robust image description and eventually better auto-annotation results. The choices of local descriptors for salient regions are multiple too, giving another direction for future work. For example, Hare (2006) described a simple colour descriptor for salient regions.

In addition, the k -means clustering method used in the quantisation step of both the “blobs” and “visual terms” representations comes with some disadvantages. Firstly, the clustering of the existing database of feature descriptors might not be sufficient to accommodate newcomers, if the data-set is to be extended. In such cases, the quantisation step, which is a very computationally expensive operation, may need to be repeated. Approaches to getting around this problem need to be developed. A possible candidate approach is through splitting and merging existing clusters which are organized in a hierarchical manner. Another problem with k -means is the choice of k , which have been chosen empirically in this work. It is interesting to develop and use an automatic method to find the optimal value, such as the G-means approach proposed by Hamerly and Elkan (2003).

8.2.2 Quality Issues with Data-Sets

For each keyword query, we have used the same number of images returned by Yahoo Image Search to build the Yahoo set, which resulted in an equalised distribution of keywords in the vocabulary. This is different from the Coral set in which some keywords occur more frequently than others. It is interesting to investigate how the difference in distribution of the vocabulary affects an auto-annotation technique as well as its evaluation. Having examined some of the disadvantages of using image data-sets like the Coral set for effective auto-annotation evaluation, we could move on to build a new benchmark set that minimises such issues. We can either reorganise the current data-set, or obtain images from other sources to build a new one. To modify the Coral set, one could try to eliminate those training images that have both extremely similar visual information and similar semantics to the test images, so that simple auto-annotation approaches such as those based on propagation do not just “luckily” pick the right ones.

Alternatively, one could download images from online image sharing websites such as Flickr, which hosts a variety of images that are more diverse from each other.

We have shown that the Corel test images can still be annotated well even when only 25% of the training information is used and it is argued that the training set contains redundant information. Choosing the best sub-set for training is worth exploring if only for computational efficiency. Training set reduction techniques (Wilson and Martinez, 2000) are of potential use for reducing the size of training sets and simultaneously filtering out the noise. We are planning to explore its application in image auto-annotation.

8.2.3 Incorporating a Statistical Model with Salient Regions

Researchers have proposed many other statistical models for image auto-annotation (section 2.2.2). The transferability of the salient region based image representation to other models are still to be investigated. For example, the CRM model (Lavrenko et al., 2003) uses the raw image feature descriptors rather than quantised ones. Directly incorporating salient region descriptors into this model may cause major computational problems, considering that the SIFT salient descriptors are of 128 dimensions and the number of salient regions in each image can reach to several thousand. On the other hand, modifying the model for automatic image region annotation is of great interest. Image regions can be chosen in a uniform way or by auto-segmentation methods, and then represented by the visual terms that fall in the region. Approximation of the probability of a word given a set of visual terms from a region is the core of the problem.

8.2.4 Non-negative Matrix Factorisation for Image Auto-annotation

In chapter 6, we have applied two versions of NMF to our experiments, the classic NMF and NMF with sparseness constraints. So far, researchers have developed many variations of the classic NMF in order to achieve better performance (Guillamet et al., 2003; Liu and Zheng, 2004; Hoyer, 2004). What kinds of enhancements to NMF are most suitable for different image tasks are still to be investigated. For example, Liu and Zheng (2004) showed that by orthonormalizing the basis vectors generated by NMF, they achieved better results in their object recognition experiments. This is probably because orthonormal bases have more discrimination power, although it might be in conflict with the non-negative purpose of NMF in the first place. Therefore, it is possible that through this approach, we can achieve better results in the experiments on semantic propagation based image auto-annotation, but probably not on object class detection, which relies on the parts-based characteristic of NMF, as orthonormalization will destroy this.

As we have mentioned, finding the optimal dimensionality of the sub-space generated by NMF is still an open and unsolved problem. Currently, we need to run experiments on

an evaluation data-set with different settings of the dimensionality to find the best one. This could be computationally very expensive. Ways to circumvent it are interesting as the same problem exists in almost all the sub-space techniques. Another shortcoming of NMF is that it does not supply a way of measuring the importance of basis vectors as SVD, in which the largest singular values correspond to the most important basis vectors. This leads to the problem that in order to calculate the basis vectors of a specific sub-space dimensionality, the decomposition process in NMF needs to be repeated entirely. On the contrary in SVD, users just need to choose the most important ones.

In the task of object class detection, we applied NMF to images that are not annotated, to find objects of the same category. A complete object recognition system should be able to relate each category to a keyword or concept. Given the ground truth annotations of the images used in our experiments, a way to relate object classes to keywords could be developed. The simplest way to find each object class a keyword is perhaps to choose the one that appears most frequently in the images that generate the segments of a particular class. In the second task, we used a very simple semantic propagation based approach for auto-annotation. The way words are propagated from training images could also be handled differently. For example, the number of training images used for propagation can be flexible, depending on how close the distances to the query image are. Besides, we plan to explore more advanced approaches that build upon the sub-space generated by NMF, for example, the linear-algebraic technique proposed by Hare et al. (2006).

8.2.5 The Image Based Feature Space Model

The development of the image based feature space (IBFS) model is at a very preliminary stage. Much work could be done to improve the model. Currently, we simply use all the training images to build the feature space. In other words, the dimensionality of the feature space equals the number of training images. Obviously, this can cause serious computational problems when it comes to a very large data-set. Techniques to reduce the dimensionality are likely to improve the performance of the model. A very simple approach would be to keep only one of the images that have extremely similar visual appearances and also have the same annotations. In this case, a mechanism to measure the visual similarity of images need to be developed, which can be either at the image level or at the region level. More advanced techniques also have potentials for this problem. For example, principal component analysis (PCA) can be utilised to find the most important training images for building the space, and remove redundant ones. The way in which Chen and Wang (2004) choose instance prototypes may also give some inspirations to this problem. They repeatedly choose an instance with the maximum diverse density (DD) value from the candidates and remove the ones that are either too similar or have small values of DD, until there are no candidates left. Therefore, the

most important instances are kept, while the repetitions and less important ones are eliminated.

Another issue regarding the IBFS model is how image segments are mapped into the space, or how the coordinates of image segments are calculated. In the work described in chapter 7, the coordinates of an image segment on a particular axis is measured by its distance to the closest segment within the training image that is represented by the axis. The cosine distance measurement is probably not the best choice. A Gaussian distribution based distance measure, which was adopted by Lavrenko et al. (2003) in their density function that is responsible for generating regional feature vectors, is perhaps more appropriate. Besides, in the experiments of section 7.2, we found that results were better when the coordinates of a segment were quantised than when the initial cosine values were used. Research into the reasons for this is needed.

8.3 The Future of Automatic Image Annotation

Since the emerging of the technique of automatic image annotation almost a decade ago, researchers from both computer vision and machine learning societies have devoted a lot of efforts to advancing it. Although very promising results have been achieved so far, research in this field still has a long way to go before this technique can be utilised in our daily life. Undoubtedly, the developments in both image description and machine learning techniques are essential to the advancement in automatic image annotation. The author regards this direction as just one side of the coin; it is a bottom-up process where images are processed and analysed based on their low-level information in order to predict the high-level objects. In the author's opinion, a top-down view of the problem also offers unique help.

For example, Enser et al. (2005) point out that one problem with the annotations generated automatically by auto-annotation techniques is the lack of richness when compared with manual annotations. They suggest employing sharable ontologies to “make explicit the relationships between the labels and concepts with which they are associated”. Ontologies can contribute to the task of image auto-annotation with useful information regarding the characteristics of concepts and the relations between them. The developments of geographic techniques bring another top-down layer of help to image auto-annotation. Today's digital cameras are featured with GPS device, and even compass. They produce photos that are associated with metadata including time stamp, location stamp and direction stamp (Naaman et al., 2003). With the help of geographic databases, the metadata can supply very valuable information about the contents of the image that was taken by the camera.

Indeed, automatic image annotation is a very challenging problem that needs the collaboration of different techniques from a variety of disciplines in order for it to achieve success.

Bibliography

- Morgan Ames and Mor Naaman. Why we tag: motivations for annotation in mobile and online media. In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 971–980. ACM, 2007.
- Kobus Barnard, Pinar Duygulu, Nando de Freitas, David Forsyth, David Blei, and Michael I. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.
- Kobus Barnard and David Forsyth. Learning the semantics of words and pictures. In *International Conference on Computer Vision*, volume 2, pages 408–415, 2001.
- Jinbo Bi, Yixin Chen, and James Z. Wang. A sparse support vector machine approach to region-based image categorization. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pages 1121–1128, Washington, DC, USA, 2005. IEEE Computer Society. ISBN 0-7695-2372-2.
- David M. Blei and Michael I. Jordan. Modeling annotated data. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 127 – 134, Toronto, Canada, 2003.
- David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.
- Miroslaw Bober. Mpeg-7 visual shape descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):716–719, June 2001.
- Peter Carbonetto, Nando de Freitas, and Kobus Barnard. A statistical model for general contextual object recognition. In *8th European Conference on Computer Vision (ECCV)*, pages 350–362, 2004.
- Gustavo Carneiro, Antoni B. Chan, Pedro J. Moreno, and Nuno Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):394–410, March 2007.

- Gustavo Carneiro and Nuno Vasconcelos. Formulating semantic image annotation as a supervised learning problem. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 163–168, 2005.
- Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines, 2001.
- O. Chapelle, P. Haffner, and V. N. Vapnik. Support vector machines for histogram-based image classification. *IEEE Transactions on Neural Networks*, 10(5):1055–1064, 1999.
- Yixin Chen, Jinbo Bi, and James Z. Wang. Miles: Multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):1931–1947, 2006. ISSN 0162-8828.
- Yixin Chen and James Z. Wang. Image categorization by learning and reasoning with regions. *The Journal of Machine Learning Research*, 5:913–939, 2004. ISSN 1533-7928.
- I.J. Cox, M.L. Miller, T.V. Minka, T.P. Papathomas, and P.N. Yianilos. The bayesian image retrieval system, pichunter: theory, implementation, and psychophysical experiments. *IEEE Transactions on Image Processing*, 9(1):20–37, January 2000.
- Ritendra Datta, Jia Li, and James Z. Wang. Content-based image retrieval: approaches and trends of the new age. In *Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, number 253 - 262, 2005.
- Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.
- Yining Deng, B. S. Manjunath, and H. Shin. Color image segmentation. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR’99*, volume 2, pages 446–451, Jun 1999.
- Thomas Deselaers, Henning Mller, Paul Clough, Hermann Ney, and Thomas M. Lehmann. The clef 2005 automatic medical image annotation task. *International Journal of Computer Vision*, 74(1):51–58, August 2007.
- S. Dumais, T. Letsche, M. Littman, and T. Landauer. Automatic cross-language retrieval using latent semantic indexing. In *AAAI Symposium on CrossLanguage Text and Speech Retrieval. American Association for Artificial Intelligence*, 1997.
- Pinar Duygulu, Kobus Barnard, Nando de Freitas, and David Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *The Seventh European Conference on Computer Vision*, pages IV:97–112, Copenhagen, Denmark, 2002.
- John P. Eakins, Jago M. Boardman, and Margaret E. Graham. Similarity retrieval of trademark images. *IEEE MultiMedia*, 5(2):53–63, 1998. ISSN 1070-986X.

- Peter G. B. Enser, Christine J. Sandom, and Paul H. Lewis. Automatic annotation of images from the practitioner perspective. In *CIVR '05: Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 497–506, 2005.
- S. L. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *Proceedings of the International Conference on Pattern Recognition (CVPR 2004)*, volume 2, pages 1002–1009, 2004.
- R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 264–271, June 2003.
- Graham Finlayson, Steven Hordley, Gerald Schaefer, and Gui Yun. Illuminant and device invariant colour using histogram equalisation. *Pattern Recognition*, 38(2):179–190, February 2005.
- Graham Finlayson and Gerald Schaefer. Colour indexing across devices and viewing conditions. In *Second International Workshop on Content-Based Multimedia Indexing*, 2001.
- Myron Flickner, Harpreet S. Sawhney, Jonathan Ashley, Qian Huang, Byron Dom, Monika Gorkani, Jim Hafner, Denis Lee, Dragutin Petkovic, David Steele, and Peter Yanker. Query by image and video content: The qbic system. *IEEE Computer*, 28(9):23–32, 1995.
- G. W. Furnas, S. Deerwester, S. T. Dumais, T. K. Landauer, R. A. Harshman, L. A. Streeter, and K. E. Lochbaum. Information retrieval using a singular value decomposition model of latent semantic structure. In *SIGIR '88: Proceedings of the 11th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 465–480, New York, NY, USA, 1988. ACM Press. ISBN 2-7061-0309-4.
- Yuli Gao, Jianping Fan, Xiangyang Xue, and Ramesh Jain. Automatic image annotation by incorporating feature hierarchy and boosting to scale up svm classifiers. In *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, pages 901–910, New York, NY, USA, 2006. ACM. ISBN 1-59593-447-2.
- David Guillaumet, Bernt Schiele, and Jordi Vitrià. Analyzing non-negative matrix factorization for image classification. In *16th International Conference on Pattern Recognition (ICPR'02)*, volume 2, pages 116–119, 2002.
- David Guillaumet and Jordi Vitrià. Evaluation of distance metrics for recognition based on non-negative matrix factorization. *Pattern Recognition Letters*, 24:1599–1605, 2003.
- David Guillaumet, Jordi Vitrià, and Bernt Schiele. Introducing a weighted non-negative matrix factorization for image classification. *Pattern Recognition Letters*, 24(14):2447–2454, 2003. ISSN 0167-8655.

- Greg Hamerly and Charles Elkan. Learning the k in k -means. In *Advances in Neural Information Processing Systems*, volume 17, 2003.
- Jonathon S. Hare. *Saliency for Image Description and Retrieval*. PhD thesis, School of Electronics and Computer Science, University of Southampton, 2006.
- Jonathon S. Hare and Paul H. Lewis. Salient regions for query by image content. In *CIVR '04: Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 317–325, 2004.
- Jonathon S. Hare and Paul H. Lewis. Content-based image retrieval using a mobile device as a novel interface. In *Storage and Retrieval Methods and Applications for Multimedia*, San Jose, California, USA, January 2005a.
- Jonathon S. Hare and Paul H. Lewis. On image retrieval using salient regions with vector-spaces and latent semantics. In *CIVR '05: Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 540–549, 2005b.
- Jonathon S. Hare and Paul H. Lewis. Saliency-based models of image content and their application to auto-annotation by semantic propagation. In *Proceedings of Multimedia and the Semantic Web / European Semantic Web Conference 2005*, 2005c.
- Jonathon S. Hare, Paul H. Lewis, Peter G. B. Enser, and Christine J. Sandom. A linear-algebraic technique with an application in semantic image retrieval. In *CIVR '06: Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 31–40, 2006.
- Atsushi Hiroike, Yoshinori Musha, Akihiro Sugimoto, and Yasuhide Mori. Visualization of information spaces to retrieve and browse image data. In *VISUAL '99: Proceedings of the Third International Conference on Visual Information and Information Systems*, pages 155–162, London, UK, 1999. Springer-Verlag.
- Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual ACM Conference on Research and Development in Information Retrieval*, pages 50–57, Berkeley, California, August 1999.
- Thomas Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1-2):177–196, January 2001.
- L Hollink, AT Schreiber, BJ Wielinga, and M Worring. Classification of user image descriptions. *International Journal of Human-Computer Studies*, 61(5):601–626, November 2004.
- Patrik O. Hoyer. Non-negative matrix factorization with sparseness constraints. *The Journal of Machine Learning Research*, 5:1457–1469, 2004.
- Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin. A practical guide to support vector classification.

- Charles E. Jacobs, Adam Finkelstein, and David H. Salesin. Fast multiresolution image querying. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 277–286, 1995.
- A.K. Jain, S. Prabhakar, and L. Hong. A multichannel approach to fingerprint classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(4): 348–359, 1999.
- J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *SIGIR 03': Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 119–126, 2003.
- I.T. Jolliffe. *Principal Component Analysis, second edition*. Springer-Verlag, New York, 2002.
- Yan Ke and Rahul Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 506–513, 2004.
- Teuvo Kohonen. A short introduction to som. Online document available at: <http://www.cis.hut.fi/projects/somtoolbox/theory/somalgorithm.shtml>.
- V. Lavrenko, R. Manmatha, and J. Jeon. A model for learning the semantics of pictures. In *Proceedings of the Seventeenth Annual Conference on Neural Information Processing Systems*, volume 16, pages 553–560, 2003.
- D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788, october 1999.
- Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems*, pages 556–562, 2001.
- J.S. Lee, S.B. Jun, and H.J. Kim. Color quantization for unconstrained images. In *International Symposium in Advances on Signal, Image Processsing, Computer Vision and Graphics*, July 1998.
- Jia Li and James Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:1075 – 1088, 2003.
- Jia Li and James Z. Wang. Real-time computerized annotation of pictures. In *Proceedings of the ACM Multimedia Conference*, pages 911–920, October 2006.
- Chih-Jen Lin. Projected gradient methods for non-negative matrix factorization. Technical Report Information and Support Service ISSTECH-95-013, Department of Computer Science, National Taiwan University, 2005.

- Weixiang Liu and Nanning Zheng. Non-negative matrix factorization based methods for object recognition. *Pattern Recognition Letters*, 25(8):893–897, 2004. ISSN 0167-8655.
- David G. Lowe. Object recognition from local scale-invariant features. In *Proc. of the International Conference on Computer Vision ICCV, Corfu*, pages 1150–1157, 1999.
- David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. ISSN 0920-5691.
- B. S. Manjunath, J. R. Ohm, V. V. Vasudevan, and A. Yamada. Color and texture descriptors. *IEEE Transactions on Circuits and System for Video Technology, Special Issue on MPEG-7*, 11(6):703–715, Jun 2001.
- B. S. Manjunath, Philippe Salembier, and Thomas Sikora, editors. *Introduction to MPEG-7: Multimedia Content Description Interface*. John Wiley & Sons, LTD, April 2002.
- Oded Maron and Tomás Lozano-Pérez. A framework for multiple-instance learning. In Michael I. Jordan, Michael J. Kearns, and Sara A. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10. The MIT Press, 1998.
- Jose M. Martinez. Mpeg-7 overview. Technical report, N6828 ISO/IEC JTC1/SC29/WG11, October 2004.
- Iain Matthews, T.F. Cootes, J.A. Bangham, S. Cox, and R. Harvey. Extraction of visual features for lipreading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):198–213, 2002.
- Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 27(10):1615–1630, 2005.
- F. Mokhtarian, S. Abbasi, and J. Kittler. Robust and efficient shape indexing through curvature scale space. In *British Machine Vision Conference*, pages 53–62, 1996.
- Farzin Mokhtarian and Mirosław Bober. *Curvature Scale Space Representation: Theory, Applications, and MPEG-7 Standardisation*. Kluwer Academic (now Springer), 2003.
- F. Monay and D. Gatica-Perez. On image auto-annotation with latent space models. In *Proceedings of the eleventh ACM international conference on Multimedia*, pages 275–278, 2003.
- Y. Mori, H. Takahashi, and R. Oka. Image-to-word transformation based on dividing and vector quantizing images with words. In *First International Workshop on Multimedia Intelligent Storage and Retrieval Management*, 1999.

- Mor Naaman, Andreas Paepcke, and Hector Garcia-Molina. From where to what: Meta-data sharing for digital photographs with geographic coordinates. In *10th International Conference on Cooperative Information Systems (COOPIS)*, 2003.
- Jia-Yu Pan, Hyung-Jeong Yang, Pinar Duygulu, and Christos Faloutsos. Automatic image captioning. In *Proceedings of the 2004 IEEE International Conference on Multimedia and Expo (ICME 2004)*, pages 1987–1990, 2004.
- Xiaojun Qi and Yutao Han. Incorporating multiple svms for automatic image annotation. *Pattern Recognition*, 40(40):728–741, February 2007.
- B. D. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, Cambridge, 1996.
- K. Rodden. How do people organise their photographs? In *BCS IRSG 21st Annual Colloquium on Information Retrieval Research*, 1999.
- Y. Rui, T. Huang, and S. Chang. Image retrieval: Current techniques, promising directions and open issues. *Journal of Visual Communication and Image Representation*, 10(4):39–62, April 1999.
- Y. Rui, T. S. Huang, and S. Mehrotra. Content-based image retrieval with relevance feedback in mars. In *Proceedings of the IEEE International Conference on Image Processing*, pages 815–818. IEEE Press, 1997.
- B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: A database and web-based tool for image annotation. Technical report, Tech. Rep. MIT-CSAIL-TR-2005-056, Massachusetts Institute of Technology, 2005.
- Bryan C. Russell, Alexei A. Efros, Josef Sivic, William T. Freeman, and Andrew Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 1605–1614, June 2006.
- N. Sebe, Q. Tian, E. Louprias, M. Lew, and T. Huang. Evaluation of salient point techniques. In *CIVR '02: Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 367–377, 2002.
- Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, pages 888–905, 2000.
- Josef Sivic, Bryan Russell, Alexei A. Efros, Andrew Zisserman, and Bill Freeman. Discovering objects and their location in images. In *International Conference on Computer Vision (ICCV)*, October 2005.
- Arnold W. M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE*

- Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000. ISSN 0162-8828.
- Michael J. Swain and Dana H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991. ISSN 0920-5691.
- Jiayu Tang, Jonathon S. Hare, and Paul H. Lewis. Image auto-annotation using a statistical model with salient regions. In *IEEE International Conference on Multimedia & Expo (ICME)*, Toronto, Ontario, Canada., 2006.
- Jiayu Tang and Paul Lewis. A study of quality issues for image auto-annotation with the corel dataset. *IEEE Transactions on Circuits and System for Video Technology*, 17:384–389, 2007a. ISSN 1051-8215.
- Jiayu Tang and Paul H. Lewis. Image auto-annotation using ‘easy’ and ‘more challenging’ training sets. In *Proceedings of 7th International Workshop on Image Analysis for Multimedia Interactive Services*, pages 121–124, Hyatt Regency, Incheon International Airport, Korea., 2006.
- Jiayu Tang and Paul H. Lewis. An image based feature space and mapping for linking regions and words. In *Proceedings of 2nd International Conference on Computer Vision Theory and Applications (VISAPP)*, pages 29–35, Barcelona, Spain, 2007b.
- Jiayu Tang and Paul H. Lewis. Using multiple segmentations for image auto-annotation. In *CIVR ’07: Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 581–586, Amsterdam, The Netherlands, 2007c. ACM Press. ISBN 978-1-59593-733-9.
- Jiayu Tang and Paul H. Lewis. Non-negative matrix factorisation for object class discovery and image auto-annotation. In *ACM International Conference on Image and Video Retrieval*, Niagara Falls, Canada, 2008.
- S. Tsuge, M. Shishibori, S. Kuroiwa, and K. Kita. Dimensionality reduction using non-negative matrix factorization for information retrieval. In *IEEE International Conference on Systems, Man, and Cybernetics*, pages 960–965, 2001.
- Mihran Tuceryan and Anil K. Jain. Texture analysis. pages 235–276, 1993.
- Matthew A. Turk and Alex P. Pentland. Face recognition using eigenfaces. In *IEEE Computer Vision and Pattern Recognition*, pages 586–591, 1991.
- University of Washington. Ground truth image database. <http://www.cs.washington.edu/research/imagedatabase/groundtruth/>, 2004.
- Juha Vesanto, Johan Himberg, Esa Alhoniemi, and Juha Parhankangas. Som toolbox for matlab 5. Technical report, Helsinki University of Technology, April 2000.

- Ville Viitaniemi and Jorma Laaksonen. Empirical investigations on benchmark tasks for automatic image annotation. In *Advances in Visual Information Systems*, volume 4781, pages 93–104. Springer, 2007.
- Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01)*, volume 1, pages 511–518, 2001.
- James Ze Wang, Jia Li, and Gio Wiederhold. SIMPLIcity: Semantics-sensitive integrated matching for picture Libraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9):947–963, 2001.
- Thijs Westerveld and Arjen P. de Vries. Experimental evaluation of a generative probabilistic image retrieval model on ‘easy’ data. In *Proceedings of SIGIR Multimedia Information Retrieval Workshop 2003*, pages 135–142, Aug 2003.
- D. Randall Wilson and Tony R. Martinez. Reduction techniques for instance-based learning algorithms. *Machine Learning*, 38(3):257–286, 2000.
- Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 267–273. ACM Press, 2003. ISBN 1-58113-646-3.
- Changbo Yang, Ming Dong, and Farshad Fotouhi. Region based image annotation through multiple-instance learning. In *MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*, pages 435–438, New York, NY, USA, 2005. ACM Press. ISBN 1-59593-044-2.
- Alexei Yavlinsky, Edward Schofield, and Stefan M. Rüger. Automated image annotation using global features and robust nonparametric density estimation. In *CIVR '05: Proceedings of the 6th ACM international conference on Image and video retrieval*, pages 507–517, 2005.